



INICIATIVA PROGRAM.AR DE LA FUNDACIÓN SADOSKY

» Propuesta curricular para la inclusión de las Ciencias de la Computación en la educación obligatoria de la Argentina



Autoría por orden alfabético:

Dabbah, Julián

Fleitas, Dolores

Garzón, Magdalena

Gómez, Marcos

Martínez, María Cecilia

Martínez López, Pablo E. 'Fidel'

Peretti, Gabriela

Con la colaboración de:

Borchardt, Mara

Labhart, Mariana



Fundación
SADOSKY

<Program.AR/>



Agradecemos a las y los docentes de educación primaria, secundaria y superior y a las y los integrantes de la iniciativa Programa.AR por su lectura crítica y sus valiosos aportes para el desarrollo de esta propuesta.

Editor: Fundación Sadosky
Coordinación editorial: Inés Roggi
Corrección y edición: Natalia Florencia Acher Lanzillotta
Diseño gráfico: Alicia Viana y Jaqueline Schaab

Cómo citar este documento: Dabbah J., Fleitas D., Garzón M., Gómez M., Martínez M.C., Martínez López P. E., Peretti, G. Propuesta curricular para la inclusión de las Ciencias de la Computación en la educación obligatoria de la Argentina. Fundación Sadosky, Buenos Aires, marzo de 2024.

La versión digital de esta publicación se encuentra en <https://curriculum.program.ar/>
 Esta publicación está disponible en acceso abierto bajo la licencia “Creative Commons Atribución-No-Comercial. Compartir igual 4.0 internacional”(CC-BY-NC-SA 4.0).
<https://creativecommons.org/licenses/by-nc-sa/4.0/>



Propuesta curricular para la inclusión de las Ciencias de la Computación en la educación obligatoria de la Argentina / Julián Dabbah ... [et al.] ; Contribuciones de Mara Borchardt ; Mariana Labhart ; Coordinación general de Inés Roggi. - 2a ed - Ciudad Autónoma de Buenos Aires : Fundación Sadosky, 2024.
 Libro digital, PDF. Archivo Digital: descarga y online

ISBN 978-987-48926-2-1

1. Educación Secundaria.
 2. Educación Primaria.
 3. Educación Inicial. I. Dabbah, Julián II. Borchardt, Mara, colab.
 - III. Labhart, Mariana, colab. IV. Roggi, Inés, coord.
- CDD 004.0712



Historial de versiones	
Versión	Cambios
VERSIÓN 2023	Fundamentos y propuesta curricular para 2do ciclo de educación primaria y secundaria (ciclo básico y ciclo orientado).
VERSIÓN 2024	Ampliación de Fundamentos e incorporación de saberes y alcances para Sala de 4 y 5 años de nivel inicial y 1er ciclo de primaria.

Prólogo

Esta propuesta curricular para la inclusión de las Ciencias de la Computación (CC) al currículo escolar¹ es el producto de las reflexiones sostenidas por el equipo de la Iniciativa Program.AR de la Fundación Sadosky e investigadoras e investigadores del campo de la Didáctica de las Ciencias de la Computación y las Ciencias de la Educación de distintas universidades nacionales a lo largo de más de una década².

En este documento se reúnen los saberes de las CC que, fruto de estas reflexiones, se consideran necesarios para ejercer una ciudadanía plena en una sociedad organizada por la tecnología computacional. La apropiación de estos saberes por parte de las y los estudiantes implica poner en juego de manera conjunta prácticas computacionales y contenidos disciplinares enmarcados en propuestas de enseñanza significativas para la comprensión del mundo contemporáneo. Es, desde este enfoque, que la Iniciativa Program.AR brinda asesoramiento desde 2016 en la actualización de los contenidos curriculares de Educación Tecnológica a las provincias de Neuquén, Córdoba, La Pampa, Santa Fe, La Rioja, la Provincia de Buenos Aires, Tucumán, Chaco y otros países como Uruguay .

En la primera parte del documento, se abordan **los fundamentos de la propuesta curricular** y se explicitan los criterios de su diseño. En la segunda parte, se presentan **los propósitos y los criterios** sobre los que se construye la propuesta curricular. En la tercera parte, se desarrolla **la estructura curricular por áreas y ejes temáticos** con dos propósitos: (I) proponer una terminología precisa que permita establecer en la escuela un lenguaje común para nombrar conceptos provenientes de las CC y (II) reflejar el modo en que estos conceptos se definen, relacionan y complejizan en la disciplina y que permita dar cuenta del espiralamiento y la profundidad de los contenidos que pueden abordarse durante toda la escolaridad obligatoria. Además, se definen cuatro prácticas computacionales para reflejar los modos de hacer de la disciplina.

En función del espacio otorgado hoy a la educación tecnológica en la escuela, esta propuesta curricular propone saberes de CC para el nivel inicial y el primer ciclo de la primaria en el marco de un área compartida con el abordaje de otras ciencias y, para el segundo ciclo de la primaria y todo el nivel secundario, un espacio curricular específico o materia.

Para la elaboración de este documento se realizó el relevamiento de propuestas curriculares de otros países y de la Argentina, y una revisión bibliográfica de los siguientes trabajos: K–12 Computer Science Framework (2016), Computer Science Curricular Guidance for Associate-Degree Transfer Programs (2017), Guide to Teaching Computer Science (2014), It is time for more critical CS education” (2020), Computer Science: A curriculum for Schools (2012), CSTA Standards (2017), ISTE Standards for Students (2016); Exploring Computer Science (2015), Núcleos de aprendizaje prioritarios de Educación Tecnológica (2011), Núcleos de Aprendizaje Prioritarios para Educación Digital, Programación y Robótica (2018)³.

Mara Borchardt
Directora Iniciativa Program.AR
Fundación Sadosky

¹ Se considera que los currículos son recortes de un campo disciplinario para ser enseñados que luego las escuelas y el cuerpo docente van reconfigurando y adaptando según sus propios saberes, los contextos sociales y políticos en donde la escuela está inserta, las condiciones institucionales y las características de su estudiantado (Terigi, 1999).

² Esta iniciativa comenzó en 2013.

³ Se puede ver el listado de todos los documentos analizados en la bibliografía.

Índice

Prólogo.....	3
Índice.....	4
Primera parte.	
Fundamentos de la propuesta.....	6
La relevancia de los saberes de las Ciencias de la Computación en el mundo contemporáneo.....	6
Qué conocimientos engloban las Ciencias de la Computación.....	7
Por qué la escuela es el lugar privilegiado para socializar estos saberes.....	8
El abordaje de las Ciencias de la Computación en el currículo argentino.....	11
Problemas que complejizan la inclusión de las CC en la educación obligatoria.....	15
Segunda parte.	
Una propuesta de Ciencias de la Computación para el sistema educativo argentino.....	19
Propósitos y enfoque de la propuesta curricular.....	21
Criterios para la selección de los contenidos.....	22
Tercera parte.	
Propuesta curricular por ciclos y niveles educativos.....	24
Organización de la disciplina escolar.....	27
Área → A. Ciudadanía y Computación.....	30
Eje → A1. Identidad digital.....	30
Eje → A2. Estrategias para un uso seguro de internet.....	31
Eje → A3. Computación y sociedad.....	32
Área → B. Programación.....	33
Eje → B1. Soluciones a problemas computacionales.....	33
Eje → B2. Representación de información en la resolución de problemas computacionales.....	34
Eje → B3. Lenguajes de programación.....	35
Área → C. Infraestructura tecnológica.....	37
Eje → C1. Organización y arquitectura de computadoras.....	37
Eje → C2. Sistemas operativos.....	38
Eje → C3. Redes e internet.....	39
Área → D. Datos e inteligencia artificial.....	40
Eje → D1. Recolección y modelado de datos.....	40
Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático.....	41
Prácticas computacionales.....	42
Saberes y alcances por nivel educativo	
Nivel Inicial y Primario.....	44
Área → A. Ciudadanía y Computación.....	45
Eje → A1. Identidad digital.....	45
Eje → A2. Estrategias para un uso seguro de internet.....	51
Eje → A3. Computación y sociedad.....	54
Área → B. Programación.....	60
Eje → B1. Soluciones a problemas computacionales.....	60
Eje → B2. Representación de información en la resolución de problemas computacionales.....	64
Eje → B3. Lenguajes de programación.....	66
Área → C. Infraestructura tecnológica.....	72

Eje → C1. Organización y arquitectura de computadoras.....	72
Eje → C2. Sistemas Operativos.....	75
Eje → C3. Redes e internet.....	78
Área → D. Datos e Inteligencia artificial.....	82
Eje → D1. Recolección y modelado de datos.....	82
Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático.....	83
Saberes y alcances por nivel educativo	
Nivel Secundario.....	87
Área → A. Ciudadanía y Computación.....	88
Eje → A1. Identidad digital.....	88
Eje → A2. Estrategias para un uso seguro de internet.....	92
Eje → A3. Computación y sociedad.....	94
Área → B. Programación.....	101
Eje → B1. Soluciones a problemas computacionales.....	101
Eje → B2. Representación de información en la resolución de problemas computacionales.....	106
Eje → B3. Lenguajes de programación.....	109
Área → C. Infraestructura tecnológica.....	117
Eje → C1. Organización y arquitectura de computadoras.....	117
Eje → C2. Sistemas operativos.....	121
Eje → C3. Redes e internet.....	123
Área → D. Datos e Inteligencia artificial.....	128
Eje → D1. Recolección y modelado de datos.....	128
Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático.....	130
Glosario de términos.....	136
Referencias bibliográficas.....	143

Primera parte.

Fundamentos de la propuesta

La relevancia de los saberes de las Ciencias de la Computación en el mundo contemporáneo

La vida contemporánea se organiza cada vez más en torno a artefactos y dispositivos computacionales. Las tarjetas de pago del transporte público, las compras mediante aplicaciones para celulares, las plataformas de series y películas, los sistemas de arbitraje para los deportes de competencia, los sistemas de seguridad por reconocimiento facial, entre otros, son todos deudores de la [tecnología computacional](#).

Se ha popularizado la idea de que usar la tecnología computacional y entenderla son equivalentes; sin embargo, no lo son. Cuando se hace referencia a “lo que las personas saben o no saben de tecnología”, muchas veces se confunde la capacidad de usar y consumir programas, aplicaciones y computadoras (incluso en calidad de prosumidor) con el conocimiento específico del funcionamiento de tales [artefactos](#) y [dispositivos computacionales](#). Esto lleva a pensar que personas (tanto adultos como también niñas, niños y adolescentes) con conocimientos sobre la utilización de redes sociales, editores de textos y videos o búsquedas en internet no necesitan saber de programación, arquitectura de computadoras ni de sistemas operativos o de la estructura de redes e internet. Sin embargo, la falta de estos saberes limita de manera creciente y significativa el empleo crítico de dichas herramientas e impide su transformación y creación.

Sin alejarnos demasiado del presente, el aislamiento obligatorio que se dispuso durante la pandemia del Covid-19 entre 2020 y 2021 aceleró los procesos de virtualización de actividades cotidianas y laborales. Esto evidenció las dificultades de algunas personas para comprender funcionamientos inesperados. Habitualmente presentaban dificultades para reconocer, describir, diferenciar e identificar en términos generales los problemas y saber si correspondían al *hardware*, al *software*, a la conexión a internet o al servicio de conectividad. Pocas contaban con el conocimiento necesario para comparar y analizar distintas opciones e implementar alternativas para resolver este tipo de problemas, y muchísimas menos tuvieron la oportunidad de dar lugar a respuestas creativas o, incluso, a pensar nuevos modos de funcionamiento.

Desde comienzos del siglo XXI y más allá de la pandemia del Covid-19, las tecnologías computacionales se han convertido en el medio de producción por excelencia. El desconocimiento de conceptos del campo limita la reflexión crítica y la toma de decisiones adecuadas de la ciudadanía sobre temas delicados que afectan la libertad, el bienestar y los derechos individuales y colectivos. No obstante, de la brecha de saberes computacionales se ha hablado bastante menos que de las brechas de acceso y uso tanto en los debates públicos como en los discursos pedagógicos especializados (Terigi, 2020). Entonces, vale preguntarse, ¿alcanza con el acceso y el uso de determinadas tecnologías o es tiempo de pensar en la necesidad de adentrarse en los modos éticos y epistemológicos que subyacen en las Ciencias de la Computación (CC)?

Si partimos de la base de que el mundo que hoy conocemos es el de la aplicación de los desarrollos de IA y, por lo tanto, la era de su implementación, aceptamos que la tarea más difícil y abstracta de la

investigación ya se ha hecho (Lee, citado en UNESCO, 2021, p. 9). Esta descripción del mundo actual como era de la implementación (frente a la de la comunicación y la información) da cuenta del carácter urgente que adquiere la enseñanza (y transmisión escolar) de esas abstracciones que configuran el modo en el que nos relacionamos con el mundo (desde los algoritmos de búsqueda, hasta los teléfonos inteligentes y chips de todo tipo, cada vez a más temprana edad).

En suma, las CC permiten reconocer la tecnología informática que se impone a otras en relación con la época y con las demandas sociales, sin caer en una visión utilitarista. Por el contrario, proponemos en esta propuesta curricular el abordaje que corresponde a un objeto del conocimiento humano en relación con las implicancias metodológicas y epistemológicas que adquiere en la construcción social. **Es decir, entendemos que las CC no se centran en el estudio de la tecnología, sino en las bases y los fundamentos sobre los que esta se monta y, por lo tanto, deberían ser parte de la educación obligatoria.**

Qué conocimientos engloban las Ciencias de la Computación

“Ciencias de la Computación” es el nombre que recibe la disciplina que estudia las bases y los fundamentos sobre los que se articula la creación y la utilización de las tecnologías computacionales. Se ocupa de los saberes necesarios para formular soluciones efectivas y sistemáticas a diversos tipos de problemas computacionales. Entre ellos y sin ser exhaustivos, retomando a Schapachnik y Bonello (2022), podemos mencionar:

- **La algoritmia:** conocimientos necesarios para formular soluciones efectivas y sistemáticas ante diversos tipos de problemas. Por ejemplo, pensemos en un sistema de posicionamiento global (GPS, por las siglas en inglés) y la sugerencia a un usuario de un camino entre todos los posibles según las condiciones de tránsito de un determinado momento.
- **La programación:** conocimientos necesarios para poder volcar esas soluciones algorítmicas a los diversos lenguajes que utilizan las computadoras. Muchas veces, y en particular cuando se habla de “llevar la programación a la escuela”, se engloba la algoritmia dentro de la programación.
- **Las estructuras de datos y las bases de datos:** dos áreas temáticas que se encargan de la forma de almacenar, gestionar y recuperar la información. Por ejemplo, hace posible que se pueda buscar velozmente un dato entre miles o millones de otros, como se hace con los buscadores de internet.
- **Las arquitecturas de computadoras:** componentes que definen los distintos tipos de computadoras y cómo estos componentes se construyen a partir de la combinación de manipulaciones sencillas de voltaje eléctrico.
- **Las redes de computadoras:** forma en que las computadoras intercambian información permitiendo el funcionamiento de internet y de todas las aplicaciones que funcionan gracias a internet, como la web, la mensajería instantánea, los juegos en línea, las transmisiones de audio y video, etcétera.
- **Teoría de la Computación:** fundamentos teóricos que marcan las diferencias entre los distintos lenguajes con sus posibilidades e imposibilidades, ventajas y desventajas, así como también otras áreas más específicas entre las que se encuentran ciertas áreas de la

matemática discreta, la teoría de la complejidad y otras como la computabilidad, que estudia qué problemas son efectivamente computables y cuáles no.

- **La inteligencia artificial⁴:** combinación de varias de las áreas previamente mencionadas para abordar problemas altamente complejos mediante mecanismos que tienen puntos en común con la cognición humana. Incluye temas como aprendizaje automático, síntesis de información, reconocimiento de voz y de imágenes, entre otros.

Ha habido diversas definiciones de la disciplina Ciencias de la Computación. Al igual que con la matemática, existe una tensión entre definirla como la ciencia de los lenguajes y los teoremas (Lu y Fletcher, 2009) o la ciencia de la resolución de problemas de cómputo (Barr y Stephenson, 2011; Isbell, 2010). Estas diferentes concepciones de la disciplina siempre tienen derivaciones en el modo en que se piensa su enseñanza.

Definida como conjunto de lenguajes y teoremas, la enseñanza de la computación se ha centrado en la escritura (y en muchas ocasiones copia) de algoritmos en lenguajes computacionales sin referencia a situaciones problemáticas (Lu y Fletcher, 2009; Zapata-Ros, 2018). Por el contrario, en cuanto disciplina que desarrolla modelos automatizables y programables para abordar problemas de cómputo, la enseñanza se ha focalizado en desafíos y situaciones que pueden ser analizadas y abordadas a partir de conceptos de las CC (Barr y Stephenson, 2011, Isbell, 2010).

En este documento, la definición de los saberes necesarios se encuadra en la perspectiva de las CC como la ciencia de la resolución de problemas de cómputo. Es desde este lugar que recobra sentido la inclusión de las Ciencias de la Computación en la escuela.

Por qué la escuela es el lugar privilegiado para socializar estos saberes

La perspectiva sobre las CC elegida tiene derivaciones a la hora de concebir una enseñanza que permita el desarrollo de una mirada crítica sobre la tecnología computacional, su adopción, sus posibilidades y sus consecuencias. Las CC entonces son vistas como una disciplina que resulta fundamental tanto para comprender un mundo cada vez más organizado y dependiente de la tecnología como para poder participar activa y soberanamente en debates actuales sobre la interacción entre las computadoras y la sociedad, y por extensión sobre la ciudadanía democrática y en pleno ejercicio de derechos y responsabilidades.

Una propuesta de enseñanza que parte desde esta concepción de las CC requiere de un enfoque pedagógico que promueva la indagación y la toma de decisiones argumentadas para fundamentar prácticas y cursos de acción. Los saberes a desarrollar deben permitir enfrentar las modificaciones continuas que se producen en el campo de la computación y brindar autonomía para discernir ventajas y desventajas de su aplicación.

⁴ Las nociones de inteligencia artificial (IA), robótica y programación no son conceptualizaciones nuevas o que conciernen exclusivamente al momento actual. Por el contrario, en los más recientes desarrollos de IA, los algoritmos relacionados con el aprendizaje profundo y las redes neuronales (que, por cierto, son los que más llaman la atención) existen desde hace más de 40 años (UNESCO, 2021, p. 9). Ha ocurrido, incluso, que dadas las bases teóricas y los modelos de trabajo, no se desarrolló la tecnología necesaria para ponerlo en práctica. Esto da cuenta de la importancia de distinguir el núcleo teórico y conceptual que aborda y problematizan las CC del alcance transversal y de época que caracteriza a la alfabetización digital y, por extensión, a la educación digital.

A la luz de una mirada humanista, la escuela es depositaria de la tradición y de la memoria, de la cultura construida y de sus procesos de producción. Esto debe ser compatible con comprender a la ciudadanía en un presente y ofrecer al estudiantado saberes que le permitan navegar e intervenir en el presente y en el futuro. Este triple rol de la institución educativa —como custodia del pasado, responsable de recapitular diversas experiencias del presente y constituir nuevas en el futuro— es lo que determina que las decisiones curriculares se encuentren siempre en tensión:

“El desafío es estructurar una idea de cultura común que pueda ser transmitida y compartida, que tome nota de las injusticias y los privilegios del pasado y que, al mismo tiempo, proponga algunas otras inclusiones que no vengan de la mano de la cultura del mercado o del individuo de autodiseño” (Dussel, 2007, p. 23).

Un ejemplo de esto es la forma en la que “lo nuevo” fue incluido dentro de la escuela (Fattore, 2007). En el campo de la educación digital, esto redundó en recuperar prácticas de computación demandadas por el mercado o por la vida social. Se instaló la idea de que usar *software* (por ejemplo, de ofimática) era el saber necesario que la escuela debía transmitir para que la ciudadanía pudiera participar del mundo digital. Esto se vuelve a expresar en la actualidad a través de la idea de que es imprescindible enseñar a usar robots o sistemas de chats de lenguaje natural (inteligencia artificial generativa). A partir de estos supuestos, se insistió con la inclusión de sistemas operativos y *software* privativo, y más recientemente la inclusión de kits de robótica, en detrimento de sostener desde lo público el acceso a la cultura libre y sus modos de producción colaborativa o de la focalización en la enseñanza de los saberes del área más que en el manejo instrumental de ciertos dispositivos.

Asimismo, la desnaturalización de patrones culturales que reproducen injusticias y desigualdades requiere de una acción positiva para que la ciudadanía en su conjunto tome conciencia y los revierta. Nuestra historia reciente está plagada de ejemplos del rol clave que cumple la escuela en esta tarea: la conciencia ecológica, los derechos de niños, niñas y adolescentes, la educación sexual integral, por mencionar algunos ejemplos de lo que se ha incluido en el currículo en las últimas décadas. La tecnología computacional no escapa a esta necesidad y su inclusión requiere revisar nuestras propias valoraciones y creencias, reflexionar sobre las expectativas de desempeño, dar cuenta de los múltiples modos en que la existencia de intereses, prejuicios y estereotipos impregnan los desarrollos tecnológicos y sus productos y visibilizar los aspectos históricos y sociales de las mujeres y las personas del colectivo LGTBQ+ en las CC.⁵

⁵ Según datos de la Unesco (2019b), las mujeres y las niñas tienen un 25 % menos de probabilidades que los hombres de aprovechar la tecnología digital para fines básicos, cuatro veces menos de saber programar computadoras y 13 veces menos de solicitar una patente tecnológica. En 2020, de acuerdo a una investigación realizada por la Fundación Sadosky, las mujeres representan el 17.9% del estudiantado en carreras informáticas. (Marino, 2023).

Un estudio realizado en escuelas primarias de Córdoba, Argentina, muestra cómo se desenvuelve el estudiantado por género frente a un mismo examen relacionado con conocimientos de programación (Martínez *et al.*, 2015, citado en Echeveste *et al.*, 2021). La variable que se observó en el estudio fue la formación vinculada a las CC recibida por las y los estudiantes. El estudio arrojó resultados contundentes: en aquellas escuelas en donde no se recibió formación específica en CC, los varones obtuvieron resultados significativamente mejores que las mujeres. En las escuelas donde estos saberes formaban parte del currículo escolar, la diferencia entre varones y mujeres fue imperceptible. Otras investigadoras han documentado las brechas digitales de género y de clase social. Por ejemplo, los trabajos de Margolis, *et al.* (2017) documentan que concebir a la enseñanza de la computación como una preparación previa para una carrera en el sector, tiende a desarrollar saberes entre quienes muestran habilidades e intereses previos signados por su posición en el campo social. Este modelo produce una selección sesgada en género, raza, y cultura y termina ofreciendo estos saberes a varones blancos de sectores socioeconómicos medios y altos.

Sin duda, la transformación digital en la que vivimos parece ser el nuevo modo de entender el mundo conocido. Simari (2013) señala que la lectura y escritura se incorpora como objeto de enseñanza en la escuela porque es una tecnología disruptiva que amplía la capacidad de cognición, de aprender y de entender el mundo que nos rodea. Esa condición “disruptiva” le permite a Simari pensar en algunos momentos clave en el desarrollo de la historia de la humanidad, desde la aparición del lenguaje como medio de comunicación hasta la máquina de Turing. Por ende, la escuela no puede (solo) enseñar a reproducir determinado saber, sino que está obligada a poner en circulación ese saber que subyace en prácticas y usos sociales cotidianos.

Finalmente, la transmisión de esa cultura común, solamente es posible si se configura un espacio disciplinario específico que dé cuenta de un objeto propio, bajo el recorte disciplinario que demanda y supone lo escolar, y que responda a unas orientaciones pedagógicas y didácticas específicas. No es el uso de las Tecnologías de la información y las comunicaciones (TIC) en todos los espacios curriculares lo que de estas aproximaciones se desprende. Las TIC pueden dar respuesta a los usos y las prácticas de artefactos, dispositivos y otras aplicaciones de la computación, en general. Sin embargo, es **el espacio curricular de las CC el que es capaz de dar cuenta de las conceptualizaciones que subyacen en sus constructos teóricos y producciones**. El uso de artefactos y dispositivos computacionales permite generar algunas intuiciones sobre las CC, pero no habilita el desarrollo de saberes para comprender, valorar, crear, e intervenir en la producción de soluciones a través de sistemas digitales. Por ello, las Ciencias de la Computación requieren ser pensadas como un cuerpo de saberes organizados, secuenciados y jerarquizados y no solamente como habilidades de uso de sistemas digitales en diferentes áreas escolares (Cuccuza, 2013; Busaniche, 2006).

Las Ciencias de la Computación como disciplina escolar

En línea con lo expuesto, entendemos a las CC como campo disciplinario que puede ser abordado en la escuela como campo/espacio curricular. Desde la perspectiva de la enseñanza para la comprensión (Boix Mansilla y Gardner, 2005) abordar una disciplina en la escuela requiere atender a cuatro dimensiones:

- **La dimensión del contenido.** Refiere a que las y los estudiantes trasciendan las creencias intuitivas y construyan una red conceptual rica y coherente. En este sentido, Pardo (2021) destaca que cada una de las distintas disciplinas posee conceptos estructurantes específicamente jerarquizados que requieren procesos de construcción singulares. Las CC como disciplina escolar requieren abordar los principios y fundamentos de conceptos claves vinculados a la programación y algoritmia, a la infraestructura tecnológica (*hardware*, sistemas operativos y redes), a la recolección, procesamiento y análisis de datos (bases de datos, ciencia de datos y aprendizaje automático), entre otros.
- **La dimensión de los métodos.** Refiere a que las y los estudiantes comprendan y pongan en práctica la forma de construir y validar conocimiento, es decir, los modos de hacer computación. Por ejemplo el reconocimiento y definición de problemas computacionales, el desarrollo y manejo de abstracciones, la creación de artefactos computacionales (algunas de las siete prácticas que identifica el marco conceptual propuesto por K-12 en Kafai, 2016).
- **La dimensión de los propósitos.** Refiere a que las y los estudiantes consideren los intereses que guían la construcción de conocimiento de la disciplina. Esta dimensión en el ámbito de las CC se vuelve indispensable dadas las consecuencias y las transformaciones que el desarrollo

computacional introdujo para el ejercicio de la ciudadanía (la identidad digital, el uso seguro de internet, la propiedad de los datos, los sesgos en los datos, la neutralidad en la red, etcétera). Se vuelve indispensable para favorecer una actitud crítica visibilizar las innovaciones digitales, los intereses que motivan la creación de *software* y *hardware*, y problematizar el control y la vigilancia en redes.

- **La dimensión de las formas de comunicación.** Refiere a que las y los estudiantes conozcan el sistema de símbolos propio de las CC y su forma de representar conocimiento. Las CC como disciplina científica tienen su propio modo de expresar, representar y comunicar datos e información.

El abordaje del objeto de las CC desde esas cuatro dimensiones permite concebirla como una disciplina escolar fundamental dado el valor y el sentido en la construcción de la ciudadanía democrática a la que se aspira.

El abordaje de las Ciencias de la Computación en el currículo argentino

En tanto la Ley 26206 de Educación Nacional de la Nación Argentina regula el ejercicio del derecho de enseñar y aprender consagrado por el artículo 14 de la Constitución Nacional argentina y los tratados internacionales incorporados a ella, abordar un objeto de estudio que se piensa en el marco de la educación obligatoria requiere pensar en términos de currículo. Es decir, dar lugar a la pregunta por el qué y para qué se enseña.

“Siendo el currículum⁶ cultura, es una cultura entre otras posibles, ¿qué contenido tiene la que unos y otros consideramos apropiada para nuestros hijos? Constituyéndose como una selección cultural que alguien realiza bajo los parámetros de una opción epistémica, política, social, cultural, de justicia, psicológica y pedagógica, ¿quiénes tienen legitimidad para proponer opciones?” (Gimeno Sacristán, 2010, p. 13).

Las decisiones educativas están motivadas por la complejidad y trascendencia pedagógica del currículo a la vez que dan cuenta de las ideologías, intereses y motivaciones de grupos y personas muy distintas y distantes que pugnan entre sí. Desde la perspectiva crítica, se entiende que el currículo no es neutral u objetivo, por el contrario, se trata de un constructo en el que confluyen factores políticos, sociales, culturales y económicos. En este sentido, el cambio curricular debe ser una oportunidad para abordar desigualdades y asegurar que todas las niñas, niños y jóvenes de una sociedad tengan acceso a una educación de calidad.

Un currículo que no aborda las desigualdades y las brechas en el aprendizaje entre diferentes grupos no puede ser considerado justo ni equitativo. Por lo tanto, el cambio curricular está justificado en las necesidades y demandas de la sociedad y los estudiantes, en la equidad y la justicia social, y en la necesidad de adaptarse a los cambios del entorno. A su vez, esta perspectiva en cuanto al modo de entender el currículo suma el enfoque emancipatorio con centro en el desarrollo de la conciencia crítica y la capacidad de las nuevas generaciones —niñas, niños, jóvenes— de transformar la sociedad.

⁶ N. de la E.: se entiende *currículum* como sinónimo de currículo.

En su desarrollo práctico, más allá de lo que figura en el texto que lo soporta, el currículo debe alcanzar y configurar ciertos derechos que superan todo análisis tecnocrático y exceden ampliamente los límites de las asignaturas escolares. Si el currículo es una construcción donde se conjuntan diferentes respuestas ante posibles opciones, donde hay que tomar decisiones de cara a un número de posibilidades, el currículo real es una posibilidad entre otras alternativas. Y el hecho de que esté vigente en un momento dado no le resta su condición de contingencia frente a su carácter no-neutro, no-universal, no-inamovible.

¿Cómo es la historia reciente de las CC en las escuelas de la Argentina?

Los desafíos de las CC en el currículo argentino tienen una historia particular y reciente que ayuda a comprender el punto en que está en la actualidad y que motivó la creación de una propuesta curricular específica para esta disciplina.

Gráfico 1: Secuencia de reglamentaciones que involucran la tecnología informática

1993	Ley Federal de Educación 24195 Art. 15: Adquisición y dominio de tecnología e informática como saberes considerados socialmente significativos
1995	Resolución CFE 39/94. En los Contenidos Básicos Comunes para la Educación General Básica se incorpora Tecnología como asignatura obligatoria con un bloque de contenidos destinado a TIC que incluye sistemas y análisis y modelado.
2006	Ley de Educación Nacional 26206 Art. 11: Desarrollo de competencias para manejar los nuevos lenguajes producidos por las TIC. Art. 88: Acceso y dominio de las TIC como contenidos curriculares indispensables.
2011	Resolución CFE 135/11 y Resolución CFE 141/11. NAP para la Educación Tecnológica. Se incluyen los procesos y medios técnicos computacionales entre las tecnologías a estudiar.
2015	Resolución CFE 263/15. "La enseñanza y el aprendizaje de la Programación es de importancia estratégica durante la escolaridad obligatoria".
2018	Resolución CFE 343/18. NAP de Educación Digital, Programación y Robótica para ser incluidos en todos los niveles del sistema educativo.

En la Argentina, desde los años 1960, la computación comenzó a tener distintos niveles de presencia en la escuela a partir de diferentes concepciones y enfoques dan cuenta de cada época.⁷ Si bien en cada momento histórico suele predominar un enfoque en función del contexto, lo cierto es que en las aulas conviven, se retroalimentan, se heredan y se disputan unos y otros de manera permanente. A pesar de esto, sirven de guía para entender las etapas que atravesó la enseñanza de computación en nuestro país.

⁷ Realizada por Cecilia Martínez en el libro *Ciencias de la Computación en la escuela* de editorial Siglo XXI (2022). Allí cita a Diego Levis con "Enseñar y aprender con informática / enseñar y aprender informática. Medios informáticos en la escuela argentina" en Levis y Cabello (Ed.), (2007). *Medios informáticos en la educación a principios del siglo XXI* (pp. 21-50). Buenos Aires: Prometeo.

En las décadas de 1970 y 1980, predominó el **enfoque técnico-operativo** que se focalizó en formar a las y los estudiantes para que operaran equipos y desarrollaran programas informáticos para el *hardware* disponible en ese momento. De alguna manera, la capacidad técnica enmarcaba el desarrollo didáctico. La decisión de introducir la computadora en la escuela estuvo orientada a preparar a los y las estudiantes para el mundo del trabajo. Luego se introdujo el lenguaje de programación Logo (el primero diseñado para enseñar a programar a niños y niñas), cuya perspectiva pedagógica fue la del aprendizaje por construcción que comienza a tomar fuerza en la Argentina avanzada la década del ochenta. La computación fue vista como una herramienta de promoción del pensamiento, y comenzó la expansión de la computadora en el hogar y la posibilidad de incluir Logo en las escuelas. De esta manera, el avance tecnológico y el acceso al equipamiento hicieron posible pensar también en una innovación pedagógica.

Durante la última década del siglo XX, los diagnósticos señalaron que, para evitar que gran parte de la población quedará excluida del acceso al mundo del trabajo y de la cultura, había que incorporar las TIC en las escuelas. En la Argentina, esta etapa coincide con la descentralización del sistema educativo nacional y el consecuente traspaso a las provincias a través de la Ley Federal de Educación (1993), que profundizó las diferencias entre provincias, y más aún en el campo de la informática educativa en prematuro desarrollo.

Ley
Federal de
Educación
(1993)

En la Ley 24195⁸, promulgada en 1993, se menciona la acelerada transformación científico-tecnológica y la necesidad de dar respuestas a estos fenómenos desde la sociedad en general y desde el sistema educativo en particular. Además, se crea el espacio curricular de Tecnología orientado a la transmisión de conocimientos y habilidades sobre el mundo artificial.⁹

En este **enfoque utilitario**, los instrumentos de la política educativa utilizados para promover la introducción de la computadora en la escuela fueron el equipamiento, la contratación y formación de docentes, y el modelo de la nueva herramienta de aprendizaje. Con el advenimiento de los grandes monopolios se impuso la lógica de enseñar los productos de ofimática en la escuela. En este enfoque, entonces, la computadora fue vista como una herramienta para apoyar alguna actividad escolar tradicional, pero no como un dispositivo para ampliar la cognición, y se solicitaba el uso de procesadores de texto, planillas de cálculos o creadores de presentaciones que reemplazaran medios analógicos.

En el inicio del nuevo milenio, aparecieron una variedad de herramientas y recursos informáticos con gran potencial educativo: algunos pensados especialmente para la educación (*softwares* educativos) y otros para enriquecer la enseñanza en prácticamente todas las materias de la escuela. A su vez, la introducción de internet generó un sesgo hacia lo exclusivamente comunicacional y el acceso a la información, lo que desdibujó los aportes conceptuales de las CC y los desarrollos didácticos en torno

⁸ Ver [Ley Federal de Educación](#).

⁹ En los *Contenidos Básicos Comunes para la Educación General Básica* del ministerio de Cultura y Educación de la Nación (Argentina, 1995) se lee: “Entre los principales aspectos que abarca la revolución científico-tecnológica se encuentran: el complejo teleinformático, determinado por la convergencia entre la microelectrónica, la informática y las telecomunicaciones; la biotecnología; los nuevos materiales; las fuentes energéticas alternativas; y el procesamiento de materiales y productos en el espacio: la robótica y la inteligencia artificial. La constante y acelerada transformación científico-tecnológica y su carácter invasivo a casi la totalidad de los aspectos de la vida diaria de las personas y las instituciones están vinculados a la aparición de fenómenos socioculturales nuevos, que requieren respuestas diferentes de la sociedad en general y del sistema educativo en particular.” (p. 226). Además, se incorporó, como espacio curricular, la Tecnología, aunque su presencia no implicó necesariamente la enseñanza de las TIC.

a su enseñanza. En esta **concepción integradora** centrada en el acceso a la información, la tecnología se consideró una herramienta, un medio para acceder y representar información que permite potenciar los aprendizajes en las diferentes disciplinas escolares. Esta concepción representó un avance sustantivo respecto del paradigma utilitario que dominó la década anterior. En la Argentina, esta etapa coincide con la promulgación de la Ley de Educación Nacional que refleja la concepción integradora.

Ley de Educación Nacional (2006)¹⁰

Se fijaron lineamientos para las escuelas en cuanto a la inclusión de contenidos y estrategias para la alfabetización digital, la enseñanza de programación y la enseñanza de tecnología:

“[La necesidad de] desarrollar las competencias necesarias para el manejo de los nuevos lenguajes producidos por las tecnologías de la información y la comunicación [...] y promover el aprendizaje de saberes científicos fundamentales para comprender y participar reflexivamente en la sociedad contemporánea” (Art. 11 del Cap. II: “Fines y objetivos de la política educativa nacional”).

A comienzos del 2010, a nivel global comienza a verse como indispensable que en todas las escuelas se pueda acceder a los diferentes saberes digitales que ya no solo incluyen usar la tecnología, sino también entender cómo funciona y poder producir nueva tecnología a partir de ella. Este enfoque, denominado por Martínez (2022) **“alfabetización digital y computacional”**, sostiene que brinda a los y las estudiantes mejores posibilidades como ciudadanos para comprender y participar en diferentes esferas de la vida social y política que los que ofrecieron los enfoques anteriores. En este marco, en la Argentina, surge la Resolución del Consejo Federal de Educación 263/15, que recupera la programación como saber prioritario.

Resolución CFE 263/15 (2015)

“La enseñanza de la Programación es de importancia estratégica durante la escolaridad obligatoria”.

Finalmente, en 2018 se aprobó la Resolución del Consejo Federal de Educación 343/18, que estableció la obligatoriedad de que todas las jurisdicciones incluyan en sus currículos una serie de saberes prioritarios vinculados a la educación digital, la programación y la robótica.

Resolución CFE 343/18 (2018)¹¹

Se aprobaron los Núcleos de aprendizaje prioritario (NAP) de Educación Digital, Programación y Robótica para todos los niveles del sistema educativo argentino. En ellos se promueve “la comprensión de los principios generales de funcionamiento de los dispositivos computarizados”.

Si bien esta normativa ha constituido un importante impulso para la efectiva introducción de saberes de las Ciencias de la Computación, ha contribuido a la confusión epistemológica respecto a cuáles son los contenidos que se engloban bajo la denominación “educación digital, programación y robótica”, su

¹⁰ Ver [Ley de Educación Nacional 26206](#).

¹¹ Ver [Resolución CFE 343/18](#).

sentido y su jerarquía. Por otro lado, otro factor que no contribuye a su implementación es que no se propone un espacio específico para abordar esos saberes.

Ejemplos de implementación en algunas jurisdicciones

En el contexto de implementación de los NAP mencionados, algunas jurisdicciones están dando pasos concretos y graduales. En el 2018, la provincia de **Neuquén**¹² aprobó un diseño curricular para el ciclo básico del nivel secundario con la inclusión de un espacio curricular de CC que pertenece a un área junto con el espacio curricular de matemáticas.

En el año 2014, la provincia de **Córdoba** puso en marcha un programa de escuelas orientadas al desarrollo de *software* emplazadas en contextos vulnerables (Escuelas ProA)¹³; y, en 2023, publicó una actualización de los contenidos de Educación tecnológica para nivel inicial, primario y secundario¹⁴. Por su parte, las provincias de **Buenos Aires** y **Corrientes** y la **Ciudad Autónoma de Buenos Aires** (CABA) en sus actualizaciones curriculares de 2023 incluyeron en nivel inicial un campo para abordar esos saberes. Adicionalmente, durante 2024, la Provincia de Buenos Aires inició la revisión del diseño curricular de nivel secundaria pensando los modos de ampliar y *aggiornar* el abordaje de la enseñanza de la tecnología computacional en la escuela. A su vez, en el contexto de la Nueva Escuela Secundaria,¹⁵ CABA incluyó contenidos de CC en la materia Tecnologías de la información en tercer y cuarto año de la educación secundaria.¹⁶

Las provincias de **Tucumán** (en 2018) y **Chaco** (en 2022) comenzaron una reforma del currículo de nivel secundario con una reformulación de los contenidos de educación tecnológica y la inclusión de proyectos y contenidos de CC de manera progresiva.¹⁷

La provincia de **La Pampa**¹⁸ trabaja, desde el 2022, en el rediseño del currículo respecto de los contenidos de Educación tecnológica para reorientarlos hacia la enseñanza de las CC de modo transversal en los niveles de educación inicial y primaria. Además, abordó el diseño de un espacio curricular específico para el nivel secundario.

En 2023, **La Rioja**, con apoyo del Consejo Federal de Inversiones, comenzó un camino de planificación de su reforma curricular; y **Santa Fe**, con apoyo del Banco de Desarrollo de América Latina (CAF), encaró un ambicioso proyecto de reformulación de contenidos para todos los niveles educativos y de actualización docente para todos los perfiles del sistema.

Problemas que complejizan la inclusión de las CC en la educación obligatoria

La enumeración anterior muestra avances, pero también da cuenta de que, en la mayoría de las jurisdicciones, no se ha logrado aún incluir significativamente los saberes de las CC en las aulas. Esto podría deberse a una combinación de factores como la indefinición de la relevancia en términos de dedicación que el abordaje de estos saberes debería tener en cada nivel educativo; la dificultad de

¹² Gobierno de la provincia de Neuquén, Consejo Provincial de Educación, [Resolución 1463/18](#).

¹³ Gobierno de Córdoba, [Escuelas ProA](#).

¹⁴ Gobierno de Córdoba, [Actualización de Educación Tecnológica](#).

¹⁵ Para más información consultar <https://buenosaires.gob.ar/educacion/escuelas/nueva-escuela-secundaria>

¹⁶ CABA, Ministerio de Educación, *Propuestas didácticas por espacios curriculares, "Tecnologías de la Información"*. ¹⁴ ["Maccione firmó convenio con la Fundación Sadosky"](#), *La Arena*, 11/07/2022.

¹⁷ Unicef, [Programa PLaNEA](#).

¹⁸ ["Maccione firmó convenio con la Fundación Sadosky"](#), *La Arena*, 11/07/2022.

definir con precisión cuáles son los saberes y prácticas adecuadas por nivel que las graduadas y los graduados debieran desarrollar; la omisión de cuál debería ser la actualización de las y los docentes en ejercicio; y la formación inicial de quienes sean responsables en las aulas de la socialización de estos saberes. Hay dos aspectos centrales en esta discusión, que han generado obstáculos: la confusión epistemológica y la determinación sobre si los saberes propios de las CC deben ser incluidos como contenidos transversales o en una materia existente en el nivel o una nueva.

Confusión epistemológica

Uno de los desafíos clave para el desarrollo curricular identificado por la UNESCO (2019c) es la **falta de claridad conceptual a nivel global** sobre las CC como disciplina académica. Algunos términos son utilizados con diferentes sentidos o incluso como sinónimos, algunos ejemplos son: TIC, la informática, la educación digital, la programación y la robótica, el pensamiento computacional¹⁹, la inteligencia artificial, la educación digital, entre otros.

El desconocimiento general de la especificidad de los saberes de las Ciencias de la Computación complejiza las discusiones y los cambios, ya que aún se confunde el desafío de su inclusión en la educación obligatoria con la integración de la tecnología digital para innovar en la enseñanza.

El concepto de **alfabetización digital**, surgido a fines de la década de los noventa de la mano de las habilidades para el siglo XXI²⁰ ha evolucionado con el tiempo. Durante la primera década del siglo XXI, se refería principalmente a saber operar artefactos computacionales (desde el encendido a utilizar paquetes de ofimática) y luego se extendió a aplicaciones que permiten no solo consumir información, sino también crearla. Posteriormente, con el surgimiento de las redes sociales, se amplió con contenidos de seguridad y privacidad en internet. De esta forma, a medida que la tecnología computacional fue mediando en la mayoría de las dimensiones de nuestra vida diaria, el concepto se fue ampliando. Actualmente, la necesidad de comprender cómo funcionan los artefactos computacionales con los que interactuamos a diario y la posibilidad de producir e intervenir la tecnología de manera ética fue cobrando fuerza como requisito para decodificar el mundo en el que vivimos²¹. Iniciada la década del 2010, esta necesidad comienza a interpelar a la escuela para que incluya los saberes que aportan las Ciencias de la Computación.

Otro concepto es el de **educación digital**, que cobró especial relevancia a partir de la Resolución 343/18, y está asociado a un conjunto de saberes entre los que puede mencionarse la ampliación de universos culturales a partir del uso de recursos digitales, la colaboración con otros/as al realizar producciones digitales, la construcción de criterios para la selección de recursos digitales, la identificación de iconografías comunes y sus funcionalidades en diferentes herramientas y contenidos

¹⁹ Nota de las y los autores: Lo que respecta a pensamiento computacional podría extenderse en una sección, sin embargo, lo hemos acotado dado que el propósito de este documento se centra en la propuesta curricular. Las habilidades que se destacan para definir lo que el pensamiento computacional involucra son: la descomposición de un problema en pasos sencillos; el reconocimiento de patrones; la abstracción; el diseño de algoritmos (entendidos como una serie de pasos que seguir), etc. Cada una de ellas despierta mucha empatía en el mundo escolar porque, desde el sentido común y desde hace siglos, en la escuela, de diferentes modos y en variedad de contextos de forma sistemática, se las ejercita. Pero es la cualidad de computacional lo que las distingue y les confiere la especificidad que se reclama a lo largo de este documento para las CC. Para ampliar este posicionamiento puede consultarse el artículo Bonello, M. .B. y Schapachnik, F. (2020) Diez preguntas frecuentes (y urgentes) sobre pensamiento computacional. Revista Virtualidad, Educación y Ciencia, 20 (11), pp. 156-167.

²⁰ Algunas de las iniciativas que han impulsado las "Habilidades para el siglo XX" son la "Alianza para el aprendizaje del siglo XXI", los Estándares de ISTE, el marco "EnGauge 21st century skills" de Meteri Group, las competencias del grupo ACT21 liderado por la Universidad de Melbourne, las Competencias clave de la OCDE, entre otros

²¹ Esta evolución se refleja en el Estudio Internacional de Alfabetización Digital y de la Información (ICILS, por sus siglas en inglés) que busca responder en qué medida están preparados las y los estudiantes para estudiar, trabajar y vivir en un mundo digital. A diferencia de estudios anteriores el ICILS mide la alfabetización digital, a partir de dos líneas: la de manejo de información digital y la de producción y transformación de la información digital vinculada a la alfabetización computacional (Fraillon *et al.*, 2014).

digitales o la construcción reflexiva de criterios para la búsqueda y la selección de las fuentes de información y recursos digitales con mediación docente, entre otros.

Como puede observarse, el término alfabetización digital no tiene límites conceptuales definidos y es de esperarse que siga evolucionando hacia un involucramiento cada vez mayor de los conceptos de las CC, hasta adentrarse en los avances en el ámbito de la inteligencia artificial y su impacto en la vida contemporánea; temas que comienzan a ser motivo de preocupación en el ámbito educativo.

La alfabetización digital que se ha promovido hasta ahora y que ha sido asimilada en algunos documentos a la educación digital (como en la Resolución 343/18), requiere de saber usar los dispositivos y artefactos computacionales. Pero limitarnos a ello en el contexto del sistema educativo supondría negar a las nuevas generaciones la posibilidad de comprender los modos de producción, funcionamiento y potencialidad que encierran dichos dispositivos y artefactos que atraviesan las prácticas y los consumos, es decir, la vida en sociedad. De la misma manera que la escuela no hace ciencia, sino que enseña acerca de la ciencia y sus modos de construcción del saber, la inclusión de las CC tiene por finalidad acercar un tipo de saber que va mucho más allá de la visión práctica y utilitarista de usar tal o cual *software* o tener habilidades frente a tal o cual *hardware*. Vale decir, las CC no encuentran su objeto disciplinario en la enseñanza del uso de programas específicos, sino en la transmisión de los saberes implicados en la construcción y el uso de los conceptos fundamentales de los que se vale²².

En síntesis, **hablar hoy de alfabetización digital implica pensar también en la alfabetización computacional**²³. A los fines de este documento, nos referiremos a la alfabetización computacional para designar a los saberes específicos que aportan las CC (la programación, la infraestructura tecnológica, la ciencia de datos e inteligencia artificial y su vinculación con la ciudadanía). Y de educación digital para designar a las habilidades requeridas como usuarios de tecnología.

Organización curricular: ¿espacio propio o transversal?

Otro de los problemas que complejizan la inclusión de las CC en la educación obligatoria es que, a diferencia de formar usuarios capaces de utilizar las TIC para enseñar, aprender y trabajar, que podía llevarse adelante de forma simultánea en todos los espacios curriculares, abordaje propio de la transversalidad, los saberes de las CC no se desarrollan sólo a través de la exposición y el uso de dispositivos digitales, sino que requieren de una enseñanza sistemática (Fraillon *et al.*, 2019). Es por eso que la alfabetización computacional está siendo considerada en muchos países y jurisdicciones argentinas como un espacio curricular propio (Bocconi *et al.*, 2016; Voogt *et al.*, 2015, Manilla *et al.*, 2014; Gobierno de la provincia de Neuquén, Consejo Provincial de Educación, Resolución 1463/18. Bases curriculares)

La educación digital, como fuera definida anteriormente, resulta un tema emergente y supone habilidades que, al igual que la lectura y la escritura, amerita un abordaje transversal. Como tal, su pretensión es promover la capacidad de utilizar programas, aplicaciones y computadoras para el aprendizaje de diferentes contenidos, lo cual no necesariamente implica conocimientos específicos de las CC y mucho menos atañe a la epistemología de la disciplina.

²² Ver [Qué conocimientos engloban las CC.](#)

²³ Resulta interesante observar que siempre que se habló de “alfabetización digital”, sin distinguir —como proponemos aquí— entre representación de la información (tecnología digital) y su procesamiento (tecnología computacional); esto produjo variaciones en el énfasis de la formación con predominancia de lo computacional en algunos momentos y de lo digital en otros (Martínez *et al.*, 2022).

Por otra parte, la teoría del discurso y práctica pedagógica de Bernstein (1973) en relación con el grado de frontera entre los contenidos y el discernimiento entre áreas de conocimiento y materias en el interior de los currículos escolares, nos permite pensar en diferentes modos de incluir las CC. En el marco de esta teoría, los currículos se pueden clasificar en débiles (también llamado “integrado” porque presenta fronteras frágiles entre las materias) y fuertes (para referirse a un currículo diferenciado y dividido en materias). Por un lado, puede abordarse la inclusión de contenidos de las CC en nivel inicial y primer ciclo de educación primaria como áreas integradas (currículo débil o integrado). Esto es posible del mismo modo que en estos niveles se incluyen contenidos específicos de la biología, la química o la literatura sin que estas disciplinas tengan en tales currículos espacios disciplinares propios. Por otro lado, la categoría de currículo fuerte permite pensar la necesidad de una materia o un espacio curricular específico para las CC a medida que se avanza en los niveles educativos dado que las materias continúan siendo las organizadoras centrales del conocimiento escolar (Dussel, 2020). A partir del segundo ciclo de la primaria y durante toda la educación secundaria, es posible abordar la enseñanza sistemática que requieren los saberes de las CC en un espacio curricular específico que aborde la computación como objeto de estudio y aporte los saberes claves necesarios para alfabetizarse computacionalmente.

Consideramos que la opción de transversalidad que habilita la Resolución 343/18 no da respuesta a la inmanencia del área de conocimiento que constituyen las CC. En este sentido, resulta indispensable insistir en que las CC constituyen un área de conocimiento que no se limita a un tema que sea plausible de abordarse en forma transversal por su carácter de impregnar todo proceso de enseñanza y de aprendizaje (Daino y Rojas, 2006). Tampoco son transversales las capacidades o habilidades que de las CC se pueden desarrollar con la pretensión de su transferencia a otros campos de conocimiento (Garzón, 2022), ya que ninguna capacidad existe en estado puro y toda capacidad se manifiesta a través de la aplicación de contenidos (Meirieu, 1990). Las disciplinas constituyen saberes poderosos, jerarquizados, articulados entre sí que no son equivalentes a modos de acercarse al conocimiento a través del sentido común, sino que requieren una introducción sistemática a modos de indagación (Dussel, 2014). Es desde estos marcos conceptuales que consideramos que transversalizar la enseñanza de una disciplina con corpus de contenidos propios conlleva el riesgo de tomar los conceptos como anecdóticos y sin la profundidad que necesitan para provocar reflexiones (Pardo, 2021).

En definitiva, la alfabetización computacional que las y los estudiantes necesitan para decodificar el mundo contemporáneo, convertirse en ciudadanas y ciudadanos plenos, creadores de nuevas soluciones y herramientas y participantes informados en los debates de actualidad, reclama un conjunto de saberes de las CC bien definido y un modo específico de abordaje en concordancia con las tradiciones de cada nivel y ciclo educativo.

Segunda parte.

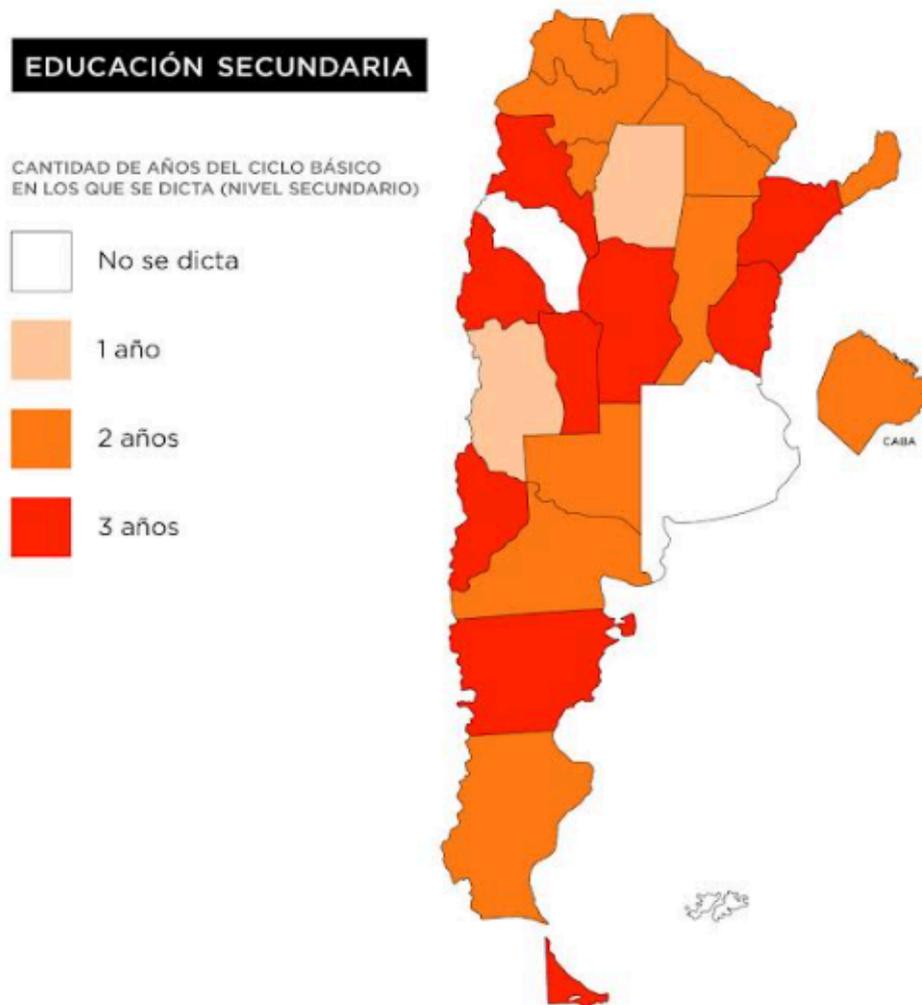
Una propuesta de Ciencias de la Computación para el sistema educativo argentino

Tras el análisis de los principales debates en torno a la inclusión de los contenidos de las CC en el currículo obligatorio y con base en diferentes aportes, concluimos que un currículo emancipador debería incluir saberes de CC en el sistema educativo obligatorio y común para todo el estudiantado. Deberá abordar con especial énfasis el rol de las tecnologías digitales y computacionales en tanto estas forman parte de la estructuración de los vínculos sociales y de la construcción de poder político y económico.

Gráfico 2. Presencia del espacio curricular de tecnología, educación tecnológica o informática en educación primaria



Gráfico 3. Presencia del espacio curricular de tecnología, educación tecnológica o informática en educación secundaria en educación secundaria



La adquisición de este conjunto de saberes requiere una diferenciación por nivel educativo y una secuenciación de complejidad progresiva. A través de las experiencias de enseñanza de las CC, se analizará y valorará el carácter histórico y geopolítico de la disciplina, lo que dará cuenta de que el desarrollo computacional depende fuertemente de los financiamientos de los Estados en favor de actividades socio-productivas, de las necesidades de solución a problemas específicos (como por ejemplo los resultantes del comercio electrónico, los sistemas de navegación automática, los efectos especiales, etcétera), de los avances tecnológicos en el campo específico, de la electrónica, y de los saberes que construyen las diferentes comunidades.

Reconocemos las tensiones sobre la inclusión de estos contenidos en la escuela (Storte, 2019), fundamentalmente, en torno a la preocupación de la comunidad educativa, en general, y de los tomadores de decisiones, en particular, por ofrecer una educación de calidad que incluya saberes relevantes para el ejercicio de la ciudadanía. Estas decisiones son siempre complejas porque seleccionar también implica dejar contenidos afuera y, además, implica dar respuesta a los marcos normativos en los que se inscribe. En tal sentido, **esta propuesta intenta acortar la brecha entre las normas educativas hoy vigentes que regulan el abordaje de la tecnología en las aulas y las realidades institucionales y de política educativa presentes en nuestro territorio.**

Esta propuesta busca contribuir a los debates que se están llevando a cabo en este momento sobre qué lugar debe tener la computadora en la escuela y qué nuevos saberes deben incorporarse en los diseños curriculares. Y pretende, especialmente, aportar claridad para la toma de decisiones en materia de política educativa en nuestro país.

A partir de estas premisas y considerando las tradiciones curriculares de nuestro país, la propuesta de distribución de saberes entre ciclos y niveles que formulamos toma en cuenta el espacio de Educación tecnológica y el hecho de que todas las jurisdicciones que han avanzado en la inclusión de contenidos de las CC han optado por la actualización de ese espacio.

Por un lado, si bien el espacio que se le otorgue a las CC siempre será una decisión jurisdiccional, sugerimos la inclusión de los contenidos y saberes específicos en el espacio de Educación tecnológica con atención al espíritu original de su formulación: dar respuesta al ambiente sociocultural con predominio de lo tecnológico en el que nos movemos en el mundo contemporáneo ²⁴ (Ferrerías y Gay*, s.f.).

“La información y las comunicaciones están en nuestra vida cotidiana. Su extendida utilización en todos los campos relacionados con la producción o la investigación justifica la necesidad de una dedicación especial para su estudio [...] lograr un conocimiento básico que permita comprender la importancia, alcance, limitaciones y perspectivas de la informática y las comunicaciones en el mundo actual” Ministerio de Cultura y Educación de la Nación, *CBC Tecnología Polimodal*, 1996, p. 6).

Por otro lado, la mayor parte de las jurisdicciones cuentan con un espacio curricular obligatorio en el ciclo básico de la educación secundaria destinado a la Educación tecnológica, lo que convierte a ese espacio en el lugar de privilegio para pensar un primer modo de institucionalización de las CC.

Propósitos y enfoque de la propuesta curricular

La inclusión de saberes de las CC a la educación obligatoria se sostiene en los siguientes propósitos generales:

- **Desnaturalizar las tecnologías computacionales:** recuperar los conceptos que las explican y que fundamentan las decisiones sobre su adopción y las consecuencias prácticas y éticas de su implementación. La comprensión de estos conceptos permite concebir las tecnologías como un desarrollo humano en donde ni los artefactos ni los dispositivos son neutrales, sino que presentan ventajas, desventajas, similitudes y diferencias con otros contemporáneos y en el tiempo.

²⁴ Dos citas son elocuentes en este sentido:

“El mundo en el que actualmente vivimos es más un mundo creado por el hombre, un mundo artificial, que un mundo natural. Casi todos los elementos que nos rodean dan testimonio del artificio humano. [...] empleo el término ‘artificial’ como el más neutro posible para indicar algo hecho por el hombre, opuesto a natural” (Simon, 1978, pp. 16-17).

“Si el mundo griego estuvo marcado por la filosofía, el romano por la jurisprudencia, el medieval por la religión, el renacentista por el arte, el moderno por la ciencia, el mundo contemporáneo lleva sin lugar a dudas la impronta de la tecnología”. (Ferrerías y Gay, *La educación tecnológica. Aportes para su implementación*, s.f.)*.

- **Focalizar la reflexión en los conceptos estables más allá del dispositivo del momento:** dado que las tecnologías computacionales se encuentran permanentemente en cambio, es relevante conocer los conceptos y los elementos estables que orientan el funcionamiento y el desarrollo de las tecnologías y que permiten enfrentar las modificaciones continuas que se producen en el campo (Zapata-Ros, 2018).
- **Valorar el impacto de las computadoras en la resolución de problemas:** centrar la enseñanza en los procesos de pensamiento para resolver problemas computacionales habilita reflexiones críticas sobre el potencial y las limitaciones de las tecnologías computacionales.
- **Relacionar los conceptos de computación con el funcionamiento de los dispositivos:** reconocer la relación de los conceptos con los dispositivos de uso cotidiano (de escritorio, celulares, tabletas, etcétera) permite comprender profundamente el funcionamiento de estos últimos, su arquitectura, sus sistemas operativos, sus aplicaciones y los programas y las formas en que pueden intercambiar información (redes e internet).
- **Comprender de qué modo la tecnología computacional afecta los derechos individuales y colectivos en la vida contemporánea** y conocer estrategias y regulaciones que permiten protegerlos y/o ejercerlos.
- **Desarrollar los saberes necesarios para crear de manera autónoma artefactos computacionales** de interés del estudiantado a partir de la participación en experiencias significativas y situadas de proyectos de programación.

Criterios para la selección de los contenidos

Esta propuesta presenta un modelo curricular con una selección de saberes fundamentales a partir del que los equipos provinciales —destinatarios clave de esta propuesta— definan el nivel de apropiación jurisdiccional de la disciplina científica a partir de cruzar el tiempo que demanda el aprendizaje de los saberes previstos en la propuesta y el tiempo de enseñanza disponible en la estructura del sistema provincial, sin perder de vista que siempre se debe garantizar enseñar “las grandes ideas” (Bell, Tymann y Yehudai, 2018; Denning, 2010): aquellos saberes que se busca que las y los estudiantes aprendan y comprendan profundamente. Para ello, desde esta propuesta los contenidos se priorizan a partir de una construcción espiralada que permitirá abordar los niveles de profundización crecientes que articulan los aprendizajes sala a sala, grado a grado, curso a curso, según las circunstancias particulares de su implementación. El proceso de diferenciación e integración progresiva contempla la necesaria flexibilidad dentro de cada ciclo y nivel. En esta propuesta de contenidos se destacan, fundamentalmente, los criterios de progresividad, coherencia y complementariedad de los aprendizajes al tiempo que se valora la diversidad de experiencias y conocimientos.

Para definir el recorte de los contenidos y las prácticas que permiten alcanzar los propósitos planteados se siguen los siguientes criterios de selección tradicionalmente utilizados en las definiciones curriculares:

- **Seleccionar los conceptos de mayor nivel de generalidad.** Se han reconocido los conceptos generales que trascienden los dispositivos y los artefactos computacionales y los problemas computacionales específicos. Al enseñar computación es fácil perderse en los

detalles técnicos y operativos en perjuicio de la visión general de la disciplina (Bell, Tymann y Yehudai, 2018). Por ejemplo: (I) para explicar qué es una computadora, en vez de enseñar la variedad de dispositivos computacionales que existen hoy en día con sus características particulares, se debería enseñar a reconocer en ellos un modelo de máquina programable al que responden todos los dispositivos y profundizar en ese modelo como objetivo del aprendizaje; (II) para dar cuenta de la idea de programa, en lugar de enseñar con un único ejemplo específico (un lenguaje en particular aplicado a un problema en particular) o con unos pocos, se podría empezar por plantear un desafío que debe ser resuelto teniendo en cuenta una serie de potencialidades y limitaciones vinculadas a la lógica de funcionamiento.

- **Identificar conceptos fundamentales que son indispensables para comprender otros.** A partir de los aportes de curriculistas y especialistas en el contenido, autores de diferentes países han ofrecido la noción de “grandes ideas” para dar cuenta de los conceptos centrales de la disciplina (Bell, Tymann y Yehudai, 2018; Denning, 2004; Goode y Chapman, 2011; Simari, 2013). Por ejemplo, las estrategias de recuperación eficiente de información explican el funcionamiento de las bases de datos, pero también el mecanismo para enrutar mensajes en internet; las nociones de modularización están presentes en la programación con bibliotecas, pero también en la organización de los circuitos que componen los dispositivos de *hardware*, en el diseño de redes y en la organización de sistemas operativos. Este criterio permite que el aprendizaje de esta materia no esté amenazado por los vertiginosos avances tecnológicos, sino que su valor consiste en brindar los fundamentos que posibilitan adaptarse a ellos sin mayores inconvenientes.
- **Elegir conceptos que promueven justicia.** Un criterio para la selección de contenidos podría ser la “justicia curricular” (Connell, 1997). Desde esta premisa y para el caso de las CC, la escuela debe ofrecer saberes que permitan atender las necesidades de conocimiento de sectores que no acceden a saberes computacionales por fuera de la escuela, para democratizar la participación ciudadana en el mundo computacional y achicar —con intención de cerrar— la brecha digital y, especialmente, la computacional. En este sentido, se abordan, por ejemplo, las normas vigentes que protegen a las infancias de delitos informáticos o acciones abusivas a través del uso de internet y la necesidad de otras, aún faltantes, que amplíen el derecho de las personas a apropiarse de la tecnología computacional en su beneficio (por ejemplo, regulaciones relacionadas con la obsolescencia programada y el derecho a reparar).
- **Asumir una perspectiva crítica con el foco en los aspectos problemáticos** de la existencia, el uso y la producción de tecnología computacional en la actualidad, entendiendo que sus aspectos positivos son los más difundidos. Este enfoque busca problematizar los efectos que tienen o pueden tener los modos de producción y distribución de la tecnología sobre las desigualdades sociales vigentes y qué estrategias pueden las ciudadanas y los ciudadanos desplegar para mitigar los impactos negativos y potenciar sus usos emancipatorios.

Tercera parte.

Propuesta curricular por ciclos y niveles educativos

La propuesta aquí presentada contempla el abordaje de las CC desde la sala de 4 y 5 años del nivel inicial hasta el último ciclo del nivel secundario. En la enumeración de contenidos, se indican los niveles en los que se sugiere su abordaje con una complejidad creciente. La **referencia a un ciclo escolar** responde a un recorrido que contempla la etapa evolutiva, los intereses y las posibilidades cognitivas de las y los estudiantes, y obedece también a una construcción espiralada de saberes desde los conceptos más cercanos y concretos hasta los más generales y abstractos. En la mayoría de los casos, **los saberes esperados para un ciclo suponen haber incorporado los saberes del ciclo anterior**. Dado que la inclusión de estos contenidos en las aulas es incipiente y seguramente su implementación sea gradual, las definiciones deberán guiarse por el hilo conductor de saberes y alcances más que por la pertenencia a un ciclo escolar.

Se organizan las CC en dos dimensiones:

1. Los contenidos que forman parte de la disciplina, organizados en áreas y ejes, bajo el título [Organización de la disciplina escolar](#),
2. Los saberes y alcances por nivel educativo en la educación obligatoria argentina, bajo el título [Saberes y alcances por nivel educativo](#)

Además, se distinguen cuatro [prácticas computacionales](#), que representan el “hacer computación”. Es en el marco de estas prácticas que los conceptos se construyen y adquieren sentido, por ello, resultan imprescindibles para poder “pensar” la computación. En este sentido, se espera que todas las propuestas de enseñanza abonen al desarrollo de estas prácticas computacionales.

1. Organización de la disciplina escolar

En este título se propone una estructura de los contenidos de las CC que se consideran relevantes para la enseñanza de esta disciplina en la escuela obligatoria. Para ello, se definen cuatro áreas que reúnen los ejes temáticos. A su vez, cada eje se contextualiza, se establece un enfoque de enseñanza y se enumeran los contenidos involucrados. El gráfico 3 sintetiza la estructura de áreas y ejes en la que se organiza la disciplina.

El orden de aparición de las áreas y ejes no supone una jerarquía ni una secuencialidad para su abordaje. En ese sentido, en el contexto de las situaciones de enseñanza que cada docente diseñe, se espera que **los contenidos englobados por las distintas áreas se aborden de manera relacionada** con una situación problemática o desafío a resolver.

Gráfico 4. Organización de la disciplina

ÁREAS	CIUDADANÍA Y COMPUTACIÓN	PROGRAMACIÓN	INFRAESTRUCTURA TECNOLÓGICA	DATOS E INTELIGENCIA ARTIFICIAL
EJES	Identidad digital	Soluciones a problemas computacionales	Organización y arquitectura de computadoras	Recolección y modelado de datos
	Estrategias para un uso seguro de internet	Representación de información en la resolución de problemas computacionales	Sistemas operativos	Ciencia de datos, inteligencia artificial y aprendizaje automático
	Computación y sociedad	Lenguajes de programación	Redes e internet	
PRÁCTICAS	Prácticas Computacionales			

Por su parte, para dar cuenta de la construcción espiralada de saberes, no todos los contenidos se abordan en todos los ciclos y niveles educativos. Mientras algunos contenidos finalizan su recorrido luego de su abordaje en dos ciclos, otros, por su complejidad, se inician recién en el ciclo básico del nivel secundario o incluso solo se abordan en el último ciclo de la educación secundaria. En el gráfico 4 es posible ver un ejemplo de esta secuenciación.

Gráfico 5. Tabla de ejemplo sobre los contenidos a abordar por ciclo educativo

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
A1.1 Convivencia en internet. Identidades digitales					
A1.2 Información pública y privada compartida en internet.					
A1.3 Huella digital					
A1.4 Vulneración de derechos y delitos informáticos. <i>Grooming</i> .					

*Este icono señala la existencia de hipervínculos que permiten recorrer el documento entre la organización disciplinaria y la organización por ciclo educativo.

2. Saberes y alcances por nivel educativo

En este título se presentan los saberes o la finalidad formativa para cada ciclo escolar enmarcados en la organización de la disciplina. A su vez, vinculado a cada contenido disciplinario, se detallan los objetivos de aprendizaje o alcances en tanto nivel de profundidad esperado.

Para facilitar la lectura y evaluar la gradualidad, se agrupa primero lo referido a los tres ciclos del [nivel inicial y primario](#) y, luego, lo que corresponde a los dos ciclos del [nivel secundario](#).

Gráfico 6. Organización por nivel educativo

Nivel inicial y primario			Nivel secundario	
Área → ÁREA A			Área → ÁREA A	
Eje → Eje 1			Eje → Eje 1	
Saberes ↓			Saberes ↓	
Inicial	Primaria, primer ciclo	Primaria, segundo ciclo	Secundaria, ciclo básico	Secundaria, ciclo orientado
Saber 1		Saber 1 Saber 2	Saber 1	Saber 1 Saber 2
Contenido → Contenido 1			Contenido → Contenido 1	
Alcances ↓			Alcances ↓	
Alcance 1 Alcance 2	Alcance 1 Alcance 2	Alcance 1	Alcance 1 Alcance 2	Alcance 1
Contenido → Contenido 2			Contenido → Contenido 2	
Alcances ↓			Alcances ↓	
Alcance 1	Alcance 1 Alcance 2	Alcance 1 Alcance 2	Alcance 1 Alcance 2	Alcance 1 Alcance 2

Organización de la disciplina escolar

Teniendo en cuenta los diferentes saberes que involucran las CC y el recorte necesario para lograr los [propósitos y enfoques](#) mencionados, se organizan los contenidos en las siguientes áreas y ejes temáticos:

Área → A. Ciudadanía y computación

Eje → A1. Identidad digital

Contenidos → A1.1 - Convivencia en internet. Identidades digitales.

Contenidos → A1.2 - Información pública y privada compartida en internet.

Contenidos → A1.3 - Huella digital.

Contenidos → A1.4 - Vulneración de derechos y delitos informáticos. *Grooming*.

Eje → A2. Estrategias para un uso seguro de internet.

Contenidos → A2.1 - Estrategias para la protección de la información privada y la identidad digital: claves seguras, permisos y sitios seguros.

Contenidos → A2.2 - Estrategias para la protección contra *software* malicioso, rastreadores, sitios falsos y *phishing*.

Eje → A3. Computación y sociedad.

Contenidos → A3.1 - Desarrollos computacionales: evolución, oportunidades y brechas, intereses y necesidades. Su influencia en la vida y los derechos de las personas. Soberanía tecnológica.

Contenidos → A3.2 - Sesgos, prejuicios y estereotipos. Filtros en el acceso a la información en internet.

Contenidos → A3.3 - El problema del poder, el control y la vigilancia en las redes. Neutralidad de la red.

Contenidos → A3.4 - Desinformación a través de internet.

Contenidos → A3.5 - Modelos de producción y distribución de *hardware* y *software*.

Contenidos → A3.6 - Propiedad de los datos. Regulación, términos y condiciones.

Contenidos → A3.7 - Computación y ambiente: obsolescencia programada y derecho a reparar, extractivismo, demanda energética e impacto ambiental.

Área → B. Programación Eje → B1. Soluciones a problemas computacionales. 

Contenidos → B1.1 - Diseño de programas: estrategias de solución, modularidad y legibilidad.

Contenidos → B1.2 - Algoritmos y estructura de datos: problemas clásicos, complejidad y recursión.

Contenidos → B1.3. Teoría de la computación: factibilidad y computabilidad.

Eje → B2. Representación de información en la resolución de problemas computacionales. 

Contenidos → B2.1 - Nociones de representación de información.

Contenidos → B2.2 - Representaciones estandarizadas (imágenes, sonido, textos y números).

Contenidos → B2.3 - Modelado: representación con datos estructurados.

Eje → Lenguajes de programación. 

Contenidos → B3.1 - La sintaxis como un aspecto de los lenguajes de programación en bloques y texto.

Contenidos → B3.2 - La semántica como el significado de los programas y sus partes en términos del problema que resuelve.

Contenidos → B3.3 - Herramientas de lenguaje de programación.

Contenidos → B3.4 - Ejecución secuencial y paralela de programas.

Contenidos → B3.5 - Compilación e interpretación: intermediación entre el código fuente y la ejecución.

Área → C. Infraestructura tecnológica Eje → C1. Organización y arquitectura de computadoras. 

Contenidos → C1.1 - La computadora como un sistema integrado por *hardware* y *software*.

Contenidos → C1.2 - Modelo de máquina programable: arquitectura von Neumann.

Contenidos → C1.3 - Componentes de *hardware*.

Contenidos → C1.4 - Ejecución a bajo nivel: lenguaje de máquina, representación binaria e impulsos eléctricos.

Eje → C2. Sistemas operativos. 

Contenidos → C2.1 - El sistema operativo como administrador de recursos e interfaz para el *hardware*.

Contenidos → C2.2 - Abstracciones del sistema operativo: archivos y sistemas de archivos, procesos, virtualización, usuarios.

Eje → C3. Redes e internet. 

Contenidos → C3.1 - Estructura y funcionamiento de redes de computadoras. Internet como una red.

Contenidos → C3.2 - Aplicaciones en internet: almacenamiento remoto, www, modelo cliente-servidor y P2P.

Contenidos → C3.3 - Mecanismos de seguridad en redes: protocolos, criptografía y anonimización.

Área → D. Datos e inteligencia artificial **Eje → D1. Recolección y modelado de datos.** 

Contenidos → D1.1 - Recolección de datos activa y pasiva.

Contenidos → D1.2 - Modelado de datos: entidades con atributos.

Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático. 

Contenidos → D2.1 - Aplicaciones de inteligencia artificial. Mecanismos generales.

Contenidos → D2.2 - Preparación, análisis y visualización de datos.

Contenidos → D2.3 - Modelos de aprendizaje automático.

Contenidos → D2.4 - Subjetividad expresada en los modelos. Sesgo algorítmico.

Área → **A. Ciudadanía y Computación**

Con la computación, la humanidad ha transformado las formas de producción de bienes y el tipo de productos existentes, la provisión de servicios y su variedad, las modalidades de esparcimiento, la vinculación, la formación y la comunicación de las personas y, por lo tanto, el modo en el que se construye y se participa de la vida contemporánea a nivel local, nacional y mundial.

La subjetividad, la organización política y las prácticas culturales son transformadas y reguladas por lo computacional. Estas transformaciones requieren ser analizadas como un producto humano no neutral, ya que responden y crean intereses, necesidades y relaciones de poder en un contexto histórico social específico.

Desde el momento en que utilizamos internet somos ciudadanas y ciudadanos digitales. En este contexto, cobra sentido abordar desde el nivel inicial pautas de convivencia en un espacio público, vincular los derechos de la infancia con su actividad en internet e iniciarse en el conocimiento de pautas de autocuidado de su privacidad e intimidad. En forma gradual, se busca que al finalizar la escolaridad obligatoria las y los estudiantes puedan ejercer una ciudadanía plena comprendiendo las funcionalidades, las potencialidades y las limitaciones de la tecnología computacional que les permita hacer uso responsable y seguro de las herramientas disponibles, participar de los debates éticos que estos cambios suponen e intervenir en la construcción del futuro de manera crítica.

Eje → A1. Identidad digital

Los espacios digitales en internet como espacios de convivencia constituyen entornos donde ejercer una ciudadanía plena y activa, con derechos y responsabilidades. En la Argentina existe un marco legal que promueve, protege y regula algunos derechos digitales (por ejemplo, el acceso, la protección de datos, la integridad sexual, la libertad y la propiedad). La actividad que realizan las personas en este espacio de convivencia, que tiene características particulares, construye una identidad digital de cada una y deja un rastro, a veces voluntario y otras involuntario.

El conocimiento de esas particularidades del entorno digital, de los derechos de las personas, de pautas de autocuidado, de modos de proteger la privacidad de cada uno, de las formas que adquiere la violencia digital y la vulneración de derechos, resulta un pilar fundamental para convivir en entornos respetuosos y formar ciudadanas y ciudadanos comprometidos y responsables.

Contenidos → A1

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
A1.1 Convivencia en internet. Identidades digitales					
A1.2 Información pública y privada compartida en internet.					
A1.3 Huella digital					
A1.4 Vulneración de derechos y delitos informáticos. <i>Grooming</i> .					

Eje → A2. Estrategias para un uso seguro de internet

Una de las necesidades originadas por el uso de tecnologías computacionales para el ejercicio de la ciudadanía reside en la adopción de medidas de seguridad frente al uso de sitios y aplicaciones en internet para prevenir delitos informáticos, robo de identidad en internet y otros daños. Tomar conciencia de los riesgos permite comprender la importancia de protegerse a través de claves seguras, permisos otorgados, y adquirir criterios para reconocer sitios inseguros, *malware*, sitios falsos y otras actividades sospechosas en la web.

Contenidos → A2

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
A2.1 Estrategias para la protección de la información privada y la identidad digital: claves seguras, permisos y sitios seguros.					
A2.2 Estrategias para la protección contra <i>software</i> malicioso, rastreadores, sitios falsos y <i>phishing</i> .					

Eje → A3. Computación y sociedad

Dadas las consecuencias y transformaciones que el desarrollo computacional introdujo para el ejercicio de la ciudadanía, se hace indispensable visibilizar el entramado en el que están insertos los artefactos computacionales a la luz de los intereses que motivan su producción y su distribución, las brechas de acceso, uso, saberes y de género y las desigualdades que derivan, la concentración de desarrollos computacionales, el poder y la soberanía o el impacto ambiental, en un contexto histórico social y regional específico.

El objetivo es que las y los estudiantes construyan una mirada más completa que se aleje de la visión simplista de considerar a la tecnología como un producto neutral y favorezca una actitud crítica que promueva y exija los derechos digitales de la ciudadanía a partir de una computación con justicia social.

Contenidos → A3

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
A3.1 Desarrollos computacionales: evolución, oportunidades y brechas, intereses y necesidades. Su influencia en la vida y los derechos de las personas. Soberanía tecnológica.					
A3.2 Sesgos, prejuicios y estereotipos. Filtros en el acceso a la información en internet.					
A3.3 El problema del poder, el control y la vigilancia en las redes. Neutralidad de la red.					
A3.4 Desinformación a través de internet.					
A3.5 Modelos de producción y distribución de <i>hardware</i> y <i>software</i> .					
A3.6 Propiedad de los datos. Regulación, términos y condiciones.					
A3.7 Computación y ambiente: obsolescencia programada y derecho a reparar, extractivismo, demanda energética e impacto ambiental.					

Área → B. Programación

El aprendizaje de la programación consiste en conocer cómo se desarrollan las soluciones de *software* para diversos problemas. El proceso de desarrollo para crear programas claros, significativos y eficientes implica elegir qué procesar, cómo procesarlo y cómo almacenar información, dividir los problemas grandes en otros más pequeños, recombinar las soluciones existentes y analizar diferentes soluciones.

Se incluye un eje sobre representación de la información en el área de programación para tener una mirada más compleja del área usualmente abordada desde las acciones. En los enfoques más comunes, los datos se utilizan, pero pocas veces se conceptualizan. Comprender cómo “aparecen” los datos al momento de programar tiene importancia para comprender el rol humano en la manipulación de estos y la incidencia de sus características en la infraestructura tecnológica.

La propuesta plantea un recorrido por la programación con un doble propósito: por un lado, en cuanto a práctica fundamental para que las y los estudiantes sean capaces de construir sus propias soluciones computacionales cada vez más complejas y, por otro, dado que el *software* es un componente imprescindible de todos los dispositivos y artefactos computacionales, conocer cómo se construye es necesario para completar los conceptos de computación trabajados en las demás áreas en cada ciclo escolar.

Eje → B1. Soluciones a problemas computacionales

La solución a problemas computacionales implica el análisis de los problemas, la definición de una estrategia y su expresión mediante la utilización de diversas técnicas y herramientas que facilitan la tarea, y que son independientes del lenguaje de programación que se utilice. Conocer y dominar técnicas para la solución de problemas computacionales, e identificarlas y analizarlas durante la construcción de programas apunta a que las y los estudiantes comprendan mejor este proceso, lo dominen desde su concepción hasta su finalización en la forma de un programa y sean capaces de construir programas legibles, reutilizables y modulares²⁵.

²⁵ Un programa es modular si está construido basándose en módulos o subprogramas. Un módulo es cada una de las partes de un programa que resuelve uno de los subproblemas en que se divide el problema complejo original. Cada uno de estos módulos tiene una tarea bien definida y algunos necesitan de otros para poder operar.

Dado que los algoritmos²⁶ son formas sofisticadas de resolver problemas, se abordan en el último nivel vinculados a nociones de teoría de la computación como la complejidad y la computabilidad.

Contenidos → B1

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
B1.1 Diseño de programas: estrategias de solución, modularidad y legibilidad.					
B1.2 Algoritmos y estructura de datos: problemas clásicos, complejidad y recursión.					
B1.3 Teoría de la computación: factibilidad y computabilidad.					

Eje → B2. Representación de información en la resolución de problemas computacionales.

La computación puede entenderse como el estudio del procesamiento de la información. Para poder ser procesada, la información debe ser primero representada dentro del sistema de cómputo, que luego manipulará dichas representaciones para lograr el procesamiento. Además, las características de la representación, tales como la eficiencia en cantidad de símbolos o la legibilidad, tienen una influencia central en las posibilidades del cómputo y habilitan aplicaciones tales como la compresión de datos o la criptografía.

Por esta razón, incorporar este eje a las prácticas de enseñanza a partir del segundo ciclo del nivel primario permite que las y los estudiantes se aproximen al concepto de representación de información y sus características y sean capaces de expresar información mediante representaciones en sus programas y trabajo con datos. Además, la reflexión desde este enfoque, permitirá identificar estas nociones fundamentales presentes en todos los procesos de cómputo.

Contenidos → B2

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

²⁶ En esta propuesta consideramos que un algoritmo es una descripción general de un proceso que permite construir una solución para todas las instancias de un problema, por ejemplo, cómo transformar una lista cualquiera de números para que quede ordenada de menor a mayor. Esta concepción es más sofisticada que la de secuencia de comandos, por eso, en esta propuesta en el nivel inicial y primario no utilizamos el término algoritmos como análogo a una secuencias de pasos para resolver tareas. Para ampliar esta información, véase la entrada del término [algoritmo](#) en el glosario de este documento.

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
B2.1 Nociones de representación de información.					
B2.2 Representaciones estandarizadas (imágenes, sonido, textos y números).					
B2.3 Modelado: representación con datos estructurados.					

Eje → B3. Lenguajes de programación

La programación es uno de los pilares de la computación, en tanto permite expresar y materializar el procesamiento de información al ejecutar los programas en una máquina. Los lenguajes de programación proveen las herramientas que permiten escribir los programas.

Solo se puede programar utilizando un lenguaje de programación. El propósito de este eje es proveer un marco general para la enseñanza de los lenguajes de programación a partir de nociones generales comunes que permitan comprender algunas herramientas centrales y transferibles entre lenguajes asociadas a los problemas que permiten resolver. Este enfoque permite correr el foco de las particularidades de un lenguaje.

La propuesta define un conjunto de herramientas de lenguajes de programación elegidas por su centralidad y generalidad. Se propone iniciar utilizando lenguajes de bloques, pues permiten focalizar en los conceptos y no lidiar con la sintaxis de los lenguajes textuales. A medida que se avanza en la escolaridad, se incorporan nuevos conceptos y herramientas y se propicia la transferencia de los conocidos a diferentes lenguajes. Este recorrido termina en los lenguajes textuales para aproximarse a las herramientas empleadas en la construcción de los programas con los que interactúan las personas en su vida contemporánea y para poder aprovechar una mayor cantidad de recursos disponibles para construir programas propios (en particular para poder utilizar los recursos de programación colaborativos como las bibliotecas).

Contenidos → B3

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
B3.1 La sintaxis como un aspecto de los lenguajes de programación en bloques y texto.					
B3.2 La semántica como el significado de los programas y sus partes en términos del problema que resuelve.					
B3.3 Herramientas de lenguaje de programación.					
B3.4 Ejecución secuencial y paralela de programas.					
B3.5 Compilación e interpretación: intermediación entre el código fuente y la ejecución.					

Área → C. Infraestructura tecnológica

Las computadoras, tal como se las conoce hoy en día, se crearon con el propósito de procesar información siguiendo programas. Se interactúa con computadoras en forma creciente: los celulares inteligentes (*smartphones*), los dispositivos de control de acceso, los cajeros automáticos, las terminales de pago con tarjetas, entre otros, son ejemplos de equipamientos que son computadoras o contienen dispositivos computacionales.

La propuesta plantea un recorrido que comienza en el nivel inicial con el reconocimiento de las características de los dispositivos computacionales que se encuentran en el entorno cotidiano. En el ciclo básico del nivel secundario, los conocimientos necesarios para poder evaluar y elegir los dispositivos computacionales que respondan a un determinado problema o necesidad, es necesario conocer cómo está construido. Para tener una mirada crítica de estos y la infraestructura asociada, se requiere, además, conocer los componentes y la arquitectura de las computadoras, identificar el rol de los sistemas operativos y comprender el funcionamiento de internet y de las redes.

Eje → C1. Organización y arquitectura de computadoras

La comprensión del funcionamiento de los dispositivos computacionales con los que interactúan las personas a diario requiere conocer cuáles son sus componentes fundamentales, cómo están organizados y cómo se relacionan entre sí para ejecutar programas. Esta visión de sistema que comienza a construirse desde el nivel inicial no solo permite una comprensión más completa, sino que al llegar al ciclo básico del nivel secundario habilitará a las y los estudiantes a diagnosticar problemas de mal funcionamiento e interpretar especificaciones técnicas para tomar un rol activo al momento de elegir las prestaciones de sus dispositivos o participar en comunidades especializadas.

Contenidos → C1

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
C1.1 La computadora como un sistema integrado por <i>hardware</i> y <i>software</i> .					
C1.2 Modelo de máquina programable: arquitectura von Neumann.					
C1.3 Componentes de <i>hardware</i> .					
C1.4 Ejecución a bajo nivel: lenguaje de máquina, representación binaria e impulsos eléctricos.					

Eje → C2. Sistemas operativos

Los sistemas operativos (SO) están presentes en los dispositivos computacionales que utilizan las personas y cumplen una función central para organizar el uso compartido de los recursos de *hardware* que permite la ejecución de los programas. El sistema operativo brinda interfaces y abstracciones que permiten que usuarias y usuarios operen sobre *software* sin preocuparse por lo que necesita el *hardware* (por ejemplo, manipular archivos y no un dispositivo de almacenamiento). Existen diferentes sistemas operativos, cada uno ofrece un sistema de archivos, un conjunto de controladores y un modo de administrar los recursos que puede ser más eficiente o no. Conocer las funcionalidades e interfaces que ofrece el sistema operativo permitirá a las y los estudiantes identificar cómo impactan estas en la experiencia de uso de los dispositivos computacionales contemporáneos.

Contenidos → C2

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
C2.1 El sistema operativo como administrador de recursos e interfaz para el <i>hardware</i> .					
C2.2 Abstracciones del sistema operativo: archivos y sistemas de archivos, procesos, virtualización, usuarios.					

Eje → C3. Redes e internet

Con frecuencia, las personas utilizan internet en la vida cotidiana sin cuestionarse su funcionamiento y estructura. Por ello, este eje propone desarmar y examinar detalladamente la red de internet para comprender cómo funciona y cómo las personas se involucran en ella. Se abordan tanto aspectos técnicos como políticos de esta red mundial de circulación de datos, desde sus componentes físicos fundamentales hasta sus diferentes capas y protocolos.

Contenidos → C3

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
C3.1 Estructura y funcionamiento de redes de computadoras. Internet como una red.					
C3.2 Aplicaciones en internet: almacenamiento remoto, www, modelo cliente-servidor y P2P.					
C3.3 Mecanismos de seguridad en redes: protocolos, criptografía y anonimización.					

Área → D. Datos e inteligencia artificial

Los datos son recolectados, almacenados, analizados y utilizados para sacar conclusiones y hacer predicciones sobre comportamientos futuros. Pero los datos recolectados siempre corresponden al pasado, su registro es imperfecto y sesgado, ya que dependen de los valores, creencias e ideas de quienes desarrollan las herramientas para recolectarlos.

El objetivo de esta área es conocer la recolección, la representación y el procesamiento de datos. Además, se abordan los procesos de análisis, consulta y transformación de los datos para realizar conclusiones. En esta misma línea, se abordan contenidos de inteligencia artificial y aprendizaje automático, prestando especial atención a sus limitaciones, la subjetividad de las personas y los datos involucrados en su desarrollo y las consecuencias de su uso.

Eje → D1. Recolección y modelado de datos

El funcionamiento de los dispositivos y artefactos computacionales contemporáneos suele involucrar la recolección de datos, a veces de manera más explícita que otras. Reconocer cuándo esto sucede y comprender los mecanismos que lo posibilitan es necesario para construir una mirada crítica y precisa sobre las experiencias de las personas, y da sustento a las reflexiones y preocupaciones que se abordan en el [eje Computación y sociedad](#). Además, poner en práctica el modelado y la recolección de datos de forma crítica será el primer paso para que las y los estudiantes analicen situaciones o fenómenos con herramientas basadas en datos.

Contenidos → D1

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
D1.1 Recolección de datos activa y pasiva.					
D1.2 Modelado de datos: entidades con atributos. Base de datos.					

Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático

Por un lado, la abundancia de datos registrados y el poder de cómputo ha convertido a las aplicaciones basadas en datos en actores cada vez más importantes para la ciudadanía. Por otro lado, los registros de datos son imperfectos y sesgados, ya que dependen de valores, creencias, contextos sociohistóricos e ideas de quienes desarrollan las herramientas para recolectarlos.

En un mundo donde la injerencia de las ciencias de datos es cada vez más amplia, cobra relevancia dotar a las y los estudiantes de herramientas que les permitan analizar y procesar datos para así poder sacar sus propias conclusiones y evaluar de forma crítica las conclusiones que otros publican. En este mismo escenario, resulta clave comprender cómo se generan las aplicaciones desarrolladas con IA, prestando especial atención a sus limitaciones, la subjetividad de las personas y los datos involucrados en su desarrollo y las consecuencias de su uso.

Contenidos → D2

Se propone abordar los contenidos que involucra este eje en los siguientes ciclos escolares:

	Inicial, sala de 4 y 5 a.	Primaria, 1.er ciclo	Primaria, 2.do ciclo	Secundaria, c. básico	Secundaria, c. orientado
D2.1 Aplicaciones de inteligencia artificial. Mecanismos generales.					
D2.2 Preparación, análisis y visualización de datos.					
D2.3 Modelos de aprendizaje automático.					
D2.4 Subjetividad expresada en los modelos. Sesgo algorítmico.					

Prácticas computacionales

Se destacan **cuatro prácticas computacionales** por ser relevantes para acercar a las y los estudiantes a los “modos de hacer” computación. Se propone incorporarlas en situaciones de enseñanza habituales. Es en el marco de estas prácticas que los conceptos incorporados en esta propuesta curricular se construyen, adquieren sentido y se organizan. Las prácticas son las siguientes.

1. Reconocer y desarrollar soluciones para problemas computacionales.
2. Crear y/o modificar artefactos computacionales.
3. Desarrollar, utilizar y abrir abstracciones.
4. Identificar y comunicar problemas de dispositivos y artefactos computacionales.

Reconocer y desarrollar soluciones para problemas computacionales

La utilización de la tecnología de la computación para la resolución de problemas se limita a determinados problemas con una serie de características específicas que denominamos “problemas computacionales”. Esta práctica involucra:

- identificar qué tipo de situaciones problemáticas pueden ser abordadas como problemas computacionales;
- conocer de forma integral los sistemas y las computadoras para definir y desarrollar una solución computacional, y
- dimensionar el impacto de la solución, tanto de las posibilidades que habilita como los nuevos problemas que puede generar.

Crear y/o modificar artefactos computacionales

Incentivar a las y los estudiantes a crear sus propios artefactos computacionales vinculados con sus intereses o que resuelvan problemas de su entorno potencia la apropiación de conceptos clave de las CC y contribuye a desarrollar la creatividad y la exploración de ideas. Es posible en las aulas crear artefactos (animaciones, videojuegos, aplicaciones móviles, sistemas web o simulaciones) combinando o reutilizando otros existentes. Esta práctica involucra:

- pensar esquemas de solución, más concretos o más abstractos, que consideren la reutilización o modificación de artefactos existentes.
- apropiarse de herramientas para el desarrollo de artefactos computacionales;

- poner a prueba y refinar artefactos computacionales, mediante un proceso deliberado e iterativo que incluye la depuración de errores (identificación y corrección) y la comparación de los resultados obtenidos con los previstos para efectuar ajustes.

Desarrollar, utilizar y abrir abstracciones

Las abstracciones, en el sentido utilizado en computación, refieren a construcciones y modelos que expresan una solución genérica.

Las abstracciones se desarrollan para crear soluciones computacionales. La utilización de abstracciones elaboradas previamente por nosotros u otras personas nos permite evitar repetir el desarrollo de soluciones de partes de un problema que ya fueron resueltas anteriormente. Las abstracciones involucran el modelado, la descomposición, la generalización y la clasificación.²⁷

Abrir una abstracción significa analizarla para comprender cómo está implementada, cuáles son sus partes y cómo contribuyen a expresar la solución correspondiente. Esto permite además de utilizar la abstracción como “caja negra” para resolver una situación problemática, comprender su lógica de construcción lo que habilita la modificación de dicha abstracción y el aprendizaje necesario para construir nuevas abstracciones.

Esta práctica computacional involucra:

- conocer las abstracciones desarrolladas por terceros;
- utilizar, en las soluciones a problemas computacionales, abstracciones predefinidas sin conocer cómo fueron desarrolladas, pero sí para qué sirven;
- comprender que el análisis de un artefacto computacional puede ser realizado en diferentes niveles de abstracción;
- reconocer que pueden existir soluciones o modelos más o menos generales.

Identificar y comunicar problemas de dispositivos y artefactos computacionales

Esta práctica se pone de manifiesto frente a problemas que pueden surgir al utilizar diferentes dispositivos y artefactos computacionales. Implica:

- identificar y describir componentes y funciones de dispositivos y artefactos computacionales utilizando la terminología adecuada para poder comunicar a otras personas sobre su funcionamiento, necesidades, problemas, etc.;
- comprender cómo se relacionan los diferentes componentes de los dispositivos y los artefactos computacionales para poder identificar el componente que podría no estar funcionando de la forma esperada;
- identificar la función de un artefacto computacional específico para poder, ante posibles fallas, investigar y buscar otros con funciones similares.²⁸

²⁷ En el Glosario se definen y ejemplifican cada uno de estos términos bajo la entrada “[Abstracciones](#)”.

²⁸ Por ejemplo, poder reemplazar un navegador específico o un programa de edición de textos por otro, si el que se está utilizando no puede llevar adelante cierta tarea.

Saberes y alcances por nivel educativo

Nivel Inicial y Primario

Área → A. Ciudadanía y Computación

Eje → A1. Identidad digital

Saberes ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

El acercamiento a internet como un espacio público de convivencia en el que participan niños/as y adultos/as; el reconocimiento de los derechos de las infancias en los entornos digitales (a la información, a la privacidad, a la identidad) y de las responsabilidades en estos espacios; la construcción de pautas de convivencia, de autocuidado y de cuidado de otras y otros y la posibilidad de identificar adultos/as de confianza para acudir ante situaciones de vulneración de los derechos en entornos digitales.

El reconocimiento del espacio público de convivencia en internet en el que ciudadanas y ciudadanos participan a través de una identidad digital, y en el cual tienen derechos específicos y la responsabilidad de intervenir con respeto y sin violencia, conociendo estrategias para detenerla y comprendiendo que la actividad en internet deja una huella.

El conocimiento de pautas de autocuidado, de la existencia de situaciones de vulneración de derechos en internet y de a quién recurrir frente a situaciones de violencia digital.

Contenidos → A1.1. Convivencia en internet. Identidades digitales

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

Reflexionar con la guía del/la docente sobre los criterios para **comunicarnos y convivir en entornos digitales**.

Conocer los vínculos y las formas de interacción en aplicaciones o juegos que utilizan internet, respetando su privacidad y la de los demás.

Reconocer que, al igual que en diferentes contextos diarios, la **convivencia en entornos digitales** requiere respetar acuerdos de convivencia ya que se vinculan con otras personas.

Reconocer los entornos digitales como **espacios de convivencia** entre personas en los que pueden participar y deben interactuar con respeto; y que como **ciudadanas y ciudadanos digitales** se tienen **derechos y responsabilidades** desde el comienzo del uso de internet (derecho a buscar, recibir y difundir información y la responsabilidad de que no resulte perjudicial u ofensiva, derecho a decidir qué información propia puede ser visible y responsabilidad no de registrar ni publicar imágenes de otras personas sin su autorización.)

Reconocer la **existencia de la identidad digital** como una construcción creada a partir del conjunto de informaciones que cada persona publica en internet y redes sociales, y que cada quien puede decidir qué reflejar en los entornos digitales.

Reconocer que las personas contribuyen a construir la **identidad digital de otra persona** a partir de la información que sobre ella publican.

Reconocer que la **identidad digital** forma parte de la vida cotidiana y de la identidad de las personas.

Reconocer que es posible falsificar y/o robar la identidad digital y a partir de ello se pueden generar situaciones de **vulneración de derechos** (por ejemplo, situaciones de *grooming*).

Reconocer que la **convivencia en entornos digitales (y no digitales) requiere del respeto de las personas, sin discriminar, dañar ni perjudicar a otros** a través de la publicación de información en internet (por ejemplo, no ejercer violencia digital a través de la divulgación de contenidos de índole sexual sin consentimiento).

		<p>Reconocer al ciberbullying como una situación de violencia digital entre pares, a través de la utilización de medios de comunicación digitales para acosar u hostigar de forma sistemática y sostenida en el tiempo a una persona, mediante ataques personales, amenazas, divulgación de información confidencial y/o falsa, entre otras estrategias.</p> <p>Reconocer las particularidades del entorno digital (el anonimato, la ausencia física que permite no ver el daño causado, el gran alcance de la información publicada, la diversidad de formatos y plataformas, la dificultad de eliminar un contenido, la asincronicidad, entre otras) que alimentan la violencia, y no minimizar su impacto, ya que lo que ocurre en internet es parte de la vida de las personas.</p> <p>Conocer estrategias para detener el ciberbullying, asumir la responsabilidad individual de no circular ni reforzar acciones de hostigamiento y conocer formas de denunciar la violencia tanto en plataformas digitales como recurriendo a un adulto de confianza.</p>
--	--	---

Contenidos → A1.2. Información pública y privada compartida en internet 

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

<p>Reflexionar con la guía del/la docente sobre las imágenes y la información que se comparten y usan.</p> <p>Construir hábitos para cuidarse y cuidar a otros/as cuando compartimos o nos comparten imágenes o información.</p> <p>Vincular las distinciones entre lo público y lo privado, el resguardo de la identidad y de la intimidad para incorporarlas al ámbito de internet.</p>	<p>Reconocer que la información o los datos que pueden compartir en aplicaciones o juegos en línea tiene que estar vinculada con el objetivo del juego o aplicación que están utilizando.</p> <p>Alertar a personas adultas de confianza ante intentos de pedido de información, datos o imágenes por parte de otro usuario/a que no se vinculen con el objetivo de la aplicación o el videojuego que están usando.</p> <p>Reconocer ámbitos digitales públicos y privados para decidir con ayuda qué datos y con quién es seguro compartir información.</p> <p>Reconocer la necesidad de cuidar la propia privacidad y respetar la privacidad de otras personas (por ejemplo, pedir permiso a la hora de publicar imágenes o videos de otras personas en redes sociales o grupos de mensajería) empleando estrategias para la protección de la información privada y la identidad digital.</p>	<p>Conceptualizar qué son los datos personales (información de cualquier tipo asociada a la identidad de una persona: nombre completo, documento, datos de localización, teléfono, la imagen personal, etc.²⁹).</p> <p>Comprender que los datos personales son privados, por qué es importante cuidarlos y saber que debe mediar el consentimiento de la persona a la que pertenecen para que otras personas u organizaciones los utilicen.</p>
--	--	---

Contenidos → A1.3. **Huella digital** 

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

²⁹ <https://www.argentina.gob.ar/justicia/convosenlaweb/situaciones/como-protger-mis-datos-personales>

--	--	<p>Reconocer que al navegar en internet se deja un rastros de información, compuesto tanto por información que generamos de manera voluntaria (poner “me gusta”, escribir un comentario a una publicación, etc.) como por el registro automático y de manera continua de nuestras acciones en internet (clics, sitios visitados, búsquedas, tiempo de permanencia, ubicación geográfica, etc.).</p>
----	----	--

Contenidos → A1.4. **Vulneración de derechos y delitos informáticos. Grooming** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Apropiarse del derecho a elegir quién puede ver el cuerpo propio o las imágenes donde aparezca en entornos virtuales y saber que ninguna persona puede exigir ni presionar a otra a que muestre su cuerpo ni causarle incomodidad, y que estas situaciones, emprendidas por un adulto, se conocen como grooming y constituyen un delito de abuso a la integridad sexual.</p> <p>Reconocer que una persona que se conecta con nosotros a través de internet puede tener un perfil falso y simular ser otra persona o tener nuestra misma edad, para emprender una situación de <i>grooming</i>.</p> <p>Comprender la importancia de recorrir a una persona adulta de confianza ante los primeros indicios de</p>

		situaciones de grooming u otra situación de violencia digital o delito informático.
--	--	---

Eje → A2. Estrategias para un uso seguro de internet 

Saberes ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

El reconocimiento de las **claves y usuarios** como herramientas para preservar la **privacidad** de las personas en internet y de la existencia de enlaces falsos o engañosos en internet.

La adopción de **medidas de seguridad para el uso de tecnologías digitales**, atendiendo a criterios de seguridad de la información, privacidad y cuestiones éticas vinculadas a la información.

Contenidos → A2.1. Estrategias para la protección de la información privada y la identidad digital: claves seguras, permisos y sitios seguros 

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

Reconocer la **existencia de claves** para el uso de diferentes aplicaciones o servicios.

Identificar en qué aplicaciones, juegos o dispositivos utilizan usuarios y/o contraseñas y que estos permiten **resguardar información que consideramos privada**.

Reconocer la necesidad de hacer un uso seguro de la tecnología para **resguardar su identidad digital** de posibles robos, alteraciones o situaciones de violencia.

Reconocer que las imágenes, videos, noticias, comentarios, gustos que se publican de forma activa en internet pueden tener distintos **grados de**

		<p>visibilidad y las personas pueden decidir, en cierta medida, esos grados de visibilidad.</p> <p>Comprender la noción de claves de acceso (contraseñas), la importancia de que estas sigan criterios de seguridad como forma de resguardar la identidad digital y evitar las consecuencias no deseadas (en particular, que cualquier persona que conozca la clave de acceso a una identidad digital pueda apropiarse de ella).</p> <p>Conocer algunos criterios de elección de claves que resulten seguras (no palabras del diccionario, no secuencias de letras consecutivas, etc.) y las maneras de utilizarlas para no alterar esa seguridad (como no compartirlas con nadie, no anotarlas en lugares obvios o fácilmente ubicables por cualquiera, no usar la misma clave para diferentes sitios, etc.).</p> <p>Reconocer cuándo un sitio utiliza un protocolo seguro (HTTPS) para proteger la información que intercambia con el usuario.</p>
--	--	--

Contenidos → A2.2. **Estrategias para la protección contra *software* malicioso, rastreadores, sitios falsos y phishing.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer situaciones en las que el contenido al que se accede (juegos, videos, imágenes, audios) no se corresponde con lo ofrecido por el vínculo de acceso. Por ejemplo, al acceder a una imagen de un video de dibujos animados para niños y niñas se accede a un sitio falso con contenidos violentos o pornografía.</p> <p>Reconocer a los/as adultos/as de confianza para pedir ayuda en caso de haber accedido a contenidos violentos, pornografía u otros contenidos inapropiados para la edad.³⁰</p>	<p>Reconocer que los juegos, las aplicaciones, los sitios o las búsquedas en internet pueden contener enlaces falsos o engañosos con la intención de generar descargas de archivos o ingresar a publicidades y sitios no solicitados.</p>	<p>Reconocer la existencia de sitios falsos tomando como indicio el nombre de la dirección web, formas de redacción del contenido y aspecto de las imágenes y certificaciones validadas (SSL).</p> <p>Reconocer aplicaciones de celular confiables tomando como indicio los permisos que solicita, las reseñas de otros usuarios, la cantidad de descargas que tiene y el autor.</p> <p>Atender a que las descargas de programas o aplicaciones de internet contengan solo lo necesario antes de ejecutar una instalación.</p>

³⁰ Como docentes, disponemos de herramientas para prevenir el acceso a contenidos inapropiados para la edad (desactivar el autoplay en las plataformas de videos, utilizar versiones kids si están disponibles).

Eje → A3. Computación y sociedad 

Saberes ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

El reconocimiento de que los **dispositivos computacionales inciden de forma distinta** en la vida de las personas y en el ambiente.

La construcción de una **mirada no neutral** sobre los artefactos y los dispositivos computacionales, su desarrollo y su distribución, así como la identificación de las consecuencias sociales y ambientales.

Contenidos → A3.1. **Desarrollos computacionales: evolución, oportunidades y brechas, intereses y necesidades. Su influencia en la vida y los derechos de las personas. Soberanía tecnológica.** 

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

Reconocer a las computadoras como **elementos artificiales** y **el rol de las personas** en su creación y distribución.

Reconocer en el **entorno cotidiano** que hay dispositivos computacionales y programas o aplicaciones que cambian las actividades cotidianas de las personas.

Reconocer que los **dispositivos computacionales** (como consolas de videojuegos o celulares) **no existieron siempre ni fueron siempre iguales** y cómo han cambiado la forma de resolver diferentes problemas.

Reconocer **situaciones** en las que el desarrollo de un **artefacto computacional** genera **nuevos problemas** (por ejemplo, limitaciones de accesibilidad para

Identificar **problemas resueltos computacionalmente** en situaciones cotidianas.

Reconocer desde una perspectiva historiográfica la forma en que han **evolucionado los artefactos y los dispositivos computacionales** de interés y uso diario (cambios en la interacción, el problema que resuelven y la forma de resolverlo).

	<p>personas con discapacidad o personas adultas mayores).</p>	<p>Reconocer que el uso de dispositivos y artefactos computacionales impacta en forma positiva o negativa en la salud de las personas, en los modos de comunicarse, entretenerse o relacionarse, en la reproducción de estereotipos (por ejemplo, los juegos destinados a niños y los destinados a niñas) y brechas de género.</p>
--	---	---

Contenidos → A3.2. **Sesgos, prejuicios y estereotipos. Filtros en el acceso a la información en internet.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer con la guía del/la docente los estereotipos de género que se reproducen en los juegos y páginas de internet que al establecer como destinatario excluyente a niños o niñas influye en la elección del juego o aplicación, reproduciendo roles de género y limitando la posibilidad de jugar y explorar libremente los entornos digitales.</p>	<p>Reconocer los indicios que transmiten roles y estereotipos de género y de cuerpos (por ejemplo, colores, objetos, formas de juego, imágenes) que promueven la elección de ciertos juegos y aplicaciones por sobre otros, limitando la posibilidad de elegir libremente cómo explorar y descubrir los entornos digitales.</p> <p>Relacionar estereotipos de la vida cotidiana³¹ que se reproducen también en los entornos digitales.</p>	<p>Reconocer que los artefactos computacionales no son objetivos, sino que reflejan la mirada de las personas y las organizaciones que los desarrollan e implementan.</p> <p>Reconocer la existencia de un recorte en la información a la que accedemos en internet (por ejemplo, en resultados de búsquedas, en las publicaciones que se ven en redes sociales, en las recomendaciones de plataformas, etc.).</p>

³¹ Se apunta a iniciar la comprensión de un recorrido sistémico en el cual los estereotipos de género y corporalidades se construyen y transmiten en muchos aspectos de la vida cotidiana (la división de tareas en el hogar, los regalos que reciben, las habilidades esperadas). El espacio digital se constituye en un entorno más donde estos estereotipos se ponen en juego.

Contenidos → A3.3. **El problema del poder, el control y la vigilancia en las redes. Neutralidad de la red.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Contenidos → A3.4. **Desinformación a través de internet.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
Tomar conciencia con la guía del/la docente de que no toda las imágenes, videos o audios que ven o escuchan en internet son verdaderos .	Tomar conciencia de que no toda la información (textos, imágenes, videos, audios) que ven, leen o escuchan en internet es verdadera .	Reconocer la facilidad con la que las personas comparten y multiplican la información que circula en internet. Reconocer que hay información que circula más que otra y que por lo general la información falsa o engañosa tiene un alto nivel de circulación . Reconocer la responsabilidad individual en la circulación de la información.

Contenidos → A3.5. Modelos de producción y distribución de *hardware* y *software*. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Reconocer que existen diferentes modos de producir <i>software</i> y <i>hardware</i>: producción abierta y colaborativa, producción cerrada.</p> <p>Reconocer que existen diferentes modos de distribuir el <i>software</i> y el <i>hardware</i>: licencias libres y licencias privativas, sus ventajas y desventajas en términos de nivel de dependencia y posibilidad de análisis y apropiación para modificar y crear artefactos computacionales.</p> <p>Reconocer cuáles de los artefactos o dispositivos computacionales que utilizamos a diario están basados en software y hardware libre, ya sea que hayan sido producidos de forma cerrada y con licencias privativas o de forma abierta y con licencia libre (por ejemplo, sistemas operativos comerciales basados en <i>software</i> libre).</p>

Contenidos → A3.6. **Propiedad de los datos. Regulación, términos y condiciones.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	Reconocer que el control del usuario sobre los datos que sube o comparte en una plataforma de internet está mediado por las empresas dueñas de los servicios que almacenan los datos (por ejemplo, cuando el usuario solicita borrar datos, es la empresa la que tiene el control de los datos físicos y decide si los borra o no, más allá de que el usuario los vea como borrados).

Contenidos → A3.7. **Computación y ambiente: obsolescencia programada y derecho a reparar, extractivismo, demanda energética e impacto ambiental.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
Tomar conciencia de que los diferentes dispositivos computacionales que se utilizan necesitan energía (electricidad, baterías, etc.) para funcionar.	Reconocer que los dispositivos computacionales son fabricados utilizando materiales cuya obtención tiene un impacto ambiental (minerales escasos, plásticos no reciclables).	Reconocer el impacto ambiental asociado a la producción y descarte de dispositivos computacionales (la basura tecnológica, extractivismo y etc.).

<p>Conocer que el uso de los dispositivos computacionales y el uso de energía eléctrica tiene impacto en el medio ambiente.</p>	<p>Generar hábitos de cuidado de los dispositivos computacionales como modo de extender su vida útil y reducir el impacto ambiental.</p>	<p>Reconocer que los dispositivos computacionales (celulares, computadoras, televisores inteligentes, etc.) se cambian en periodos de tiempo relativamente cortos por motivos de diseño por parte de los fabricantes (obsolescencia programada³²).</p> <p>Reconocer el impacto ambiental de la demand de infraestructura física y energética de los servicios de internet (como redes sociales, buscadores, servicios de mensajería, videojuegos en línea, etc.).</p>
--	---	---

³² La obsolescencia programada es la vida útil que le da una fábrica o empresa a un producto. Cuando pasa este periodo el producto se vuelve inservible. La mayoría de los productos están “programados para morir”. Muchas veces cuando estos dispositivos dejan de funcionar es más económico comprar uno nuevo que reparar el que ya tenemos (ya sea por el costo de los repuestos o porque ya salió un modelo más actualizado). La obsolescencia programada se creó para que el consumidor se viera obligado a adquirir un producto nuevo igual o similar y cubrir esa necesidad. Esto influye mucho en el desarrollo de la economía, ya que asegura una gran demanda, donde las empresas tienen más beneficios y una continua oferta. ([Ortega, O. Programa de Renovación y Actualización de máquinas \(RAM\). UNC 2021](#)).

Área → B. Programación 

Eje → B1. Soluciones a problemas computacionales 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>La resolución de problemas computacionales mediante la creación de programas con bloques icónicos, identificando elementos repetitivos en el problema y utilizando las herramientas específicas de lenguajes de programación (secuencia, repetición y comandos primitivos).</p>	<p>La resolución de problemas computacionales mediante la creación de programas con bloques, identificando elementos repetitivos en el problema y diferenciando características que se mantienen fijas de las que varían, en vinculación con las herramientas específicas de lenguajes de programación (secuencia, repetición, comandos primitivos, alternativa condicional y eventos).</p>	<p>La resolución de problemas computacionales a través del diseño de estrategias que permitan estructurar y construir programas, identificando elementos repetitivos y regularidades en el problema, expresando acciones mediante división en subtareas y expresando estas como procedimientos con denominaciones adecuadas, siempre en vinculación con las herramientas específicas de lenguajes de programación.</p> <p>La comprensión de que el programa producido es una forma de comunicar la solución propuesta a personas y máquinas.</p>

Contenidos → B1.1. **Diseño de programas: estrategias de solución, modularidad y legibilidad.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer si el programa resuelve el objetivo del problema planteado.</p> <p>Entender a los programas como una secuencia de comandos³³.</p> <p>Reconocer que dos programas compuestos por distintas secuencias de comandos pueden alcanzar un mismo objetivo.</p> <p>Reconocer los errores en un programa y proponer cambios o nuevos programas para resolver el problema.</p> <p>Identificar patrones en el contexto o escenario de un desafío de programación para utilizar repetición en vez de secuencias repetidas de comandos primitivos.</p>	<p>Reconocer qué características de un problema se mantienen fijas y cuáles varían y por lo tanto requieren el uso de la alternativa condicional para expresar diferentes comportamientos para la solución del problema³⁴.</p> <p>Reconocer si el programa resuelve el objetivo del problema planteado, teniendo en cuenta sus variaciones.</p> <p>Reconocer que pueden existir diferentes programas que resuelvan un mismo problema.</p> <p>Identificar regularidades en un programa y expresarlas a través de repeticiones.</p> <p>Reconocer en la descripción de un problema, las relaciones entre las acciones del usuario y el comportamiento esperado del programa, y expresarlas a través de eventos.</p>	<p>Identificar, diferenciar y comprender situaciones problemáticas donde haya intervenido una solución computacional o parte de ella y, en particular, sistemas que incluyan componentes de <i>software</i> para cómputo.</p> <p>Elaborar, a partir de un problema enunciado claramente y con límites precisos, una estrategia de solución que pueda ser implementada con las herramientas provistas por el lenguaje de programación.</p> <p>Identificar características relevantes, comportamientos repetidos y características fijas y variables de una situación problemática claramente enunciada y con límites precisos para elaborar una estrategia de solución a partir de descomponerla en términos de subproblemas(técnica de división en subproblemas).</p> <p>Utilizar la técnica de división de un problema en subproblemas tanto para pensar y diseñar estrategias de solución a problemas como para programar mediante el uso de procedimientos³⁵ asignándoles</p>

³³ Para conocer los motivos por lo que se decide utilizar la noción de secuencia de comandos en lugar de algoritmos, véase la aparición del término [Algoritmo](#) en el glosario al final de este documento.

³⁴ Por ejemplo: en un desafío, la posición del objetivo es fija mientras que la aparición de un obstáculo varía y requiere que la solución prevea recorridos diferentes

³⁵ Ver el eje B3. Lenguajes de programación.

		<p>una denominación representativa en términos del dominio del problema³⁶.</p> <p>Reflexionar sobre el hecho de que un programa es una manera de comunicar soluciones a problemas computacionales entre personas que programan, y que, al mismo tiempo, debe servir para hacer funcionar a una máquina que solucione el problema al ejecutar el código.</p> <p>Reconocer la necesidad de conservar, acceder y modificar información (por ejemplo, el puntaje actual en un juego).</p> <p>Reconocer la existencia de soluciones previas que pueden ser utilizadas como subtareas en una solución propia sin conocer cómo están implementadas.</p> <p>Identificar regularidades en un programa y expresarlas a través de las herramientas adecuadas del lenguaje (por ejemplo, repeticiones y procedimientos).</p> <p>Construir la noción de un programa correcto, teniendo en cuenta si los resultados obtenidos al ejecutarlo son los esperados con respecto al problema.</p>
--	--	--

³⁶ El dominio de un problema son los elementos que integran el problema y las relaciones que existen entre ellos en el nivel del problema. Por ejemplo, si se está programando una aplicación bancaria, el dominio habla de “cuentas bancarias”, de “depositar en una cuenta”, de “extraer de una cuenta”, de “saldo de una cuenta”, etc. Luego, según cómo se elige representar las cuentas, las operaciones se implementan en términos de la representación. Entonces, si para una cuenta se elige guardar dos datos (tipo y número), la operación de depositar va a incrementar el número, la de extraer verificará si el número es mayor que cero en una caja de ahorros, o mayor que cierto “descubierto” en una cuenta corriente, y restará, etc.

Contenidos → B1.2. **Algoritmos y estructura de datos: problemas clásicos, complejidad y recursión.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Contenidos → B1.3. **Teoría de la computación: factibilidad y computabilidad.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Eje → **B2. Representación de información en la resolución de problemas computacionales** 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	El reconocimiento de la representación de información en diferentes situaciones y contextos para utilizar herramientas de lenguajes de programación que representen información (como sensores ³⁷).

Contenidos → **B2.1. Nociones de representación de información.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	Identificar situaciones en las que se representan entidades mediante datos ³⁸ .

³⁷ Ver el eje B3 de Lenguajes de programación.

³⁸ Por ejemplo, utilizar un número 1 para representar una manzana y un número 2 para representar una banana, u otras representaciones sencillas, tales como representar puntos cardinales con números, estados del mar con banderas de colores, colores en los semáforos, etc.

		<p>Reconocer que la representación de una entidad es arbitraria y la definen las personas en un contexto³⁹.</p> <p>Reconocer las herramientas de lenguajes de programación que permiten expresar información (sensores y expresiones literales⁴⁰) y utilizarlas en operaciones sencillas (detección de condiciones simples, operaciones aritméticas simples).</p>
--	--	--

Contenidos → B2.2. Representaciones estandarizadas (imágenes, sonido, texto y números). 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	Conocer la forma en que se representan videos, imágenes y textos en los artefactos computacionales ⁴¹ .

³⁹ Por ejemplo, los antiguos romanos representaban los números de una forma y nosotros de otra.

⁴⁰ Ver el eje B3. Lenguajes de programación.

⁴¹ Por ejemplo, un video representado como una secuencia de imágenes, una imagen como una cuadrícula de píxeles y que cada píxel se representa con tres números, un texto representado como secuencia de caracteres, y cada carácter representado como un número.

Contenidos → B2.3. Modelado: representación con datos estructurados. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Eje → B3. Lenguajes de programación 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
La comprensión de conceptos generales de lenguajes de programación y la construcción de programas utilizando las herramientas ⁴² propuestas para este ciclo (comandos primitivos, secuencia, repetición simple).	La comprensión de conceptos generales de lenguajes de programación y la construcción de programas sumando las herramientas propuestas para este ciclo (alternativa condicional, eventos).	La comprensión de conceptos generales de lenguajes de programación y la construcción de programas sumando las herramientas propuestas para este ciclo (procedimientos, variables, listas y expresiones con sus tipos).

⁴² Ver el contenido [B3.3 Herramientas de lenguajes de programación](#) por definiciones de todas estas herramientas.

Contenidos → B3.1. **La sintaxis como un aspecto de los lenguajes de programación en bloques y texto.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
Identificar la necesidad de contar con un lenguaje de programación , en este nivel será icónico, para crear programas que la computadora pueda ejecutar.	Reconocer que el lenguaje de programación define diferentes reglas que nos permiten crear programas que pueden ejecutar las computadoras.	Reconocer que existen reglas para la combinación de bloques que determinan cuáles son las combinaciones válidas.

Contenidos → B3.2. **La semántica como el significado de los programas y sus partes en términos del problema que resuelve.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
Asociar en un programa de lenguaje icónico cada uno de los bloques con su efecto en la ejecución .	Reconocer que los comandos describen acciones. Reconocer a las condiciones como preguntas que se pueden responder por sí o por no, y que impactan en el comportamiento del programa. Anticipar , sin necesidad de ejecutar, qué modificaciones del escenario produce (qué hace) una	Construir el significado de un procedimiento como modificación que se produce en el escenario al ejecutarlo, distinguiéndolo de la secuencia de pasos involucrados en tal modificación. ⁴³ Comprender que la definición de un procedimiento incorpora un nuevo comando al lenguaje.

⁴³ Por ejemplo, el contraste entre “ir a la puerta” y la secuencia “mover 4 pasos a la derecha, y luego mover 3 pasos adelante”.

	<p>secuencia de comandos en un entorno de enseñanza para la programación .</p>	<p>Reconocer la diferencia entre comandos y expresiones, en tanto descripciones de acciones y datos, respectivamente.</p> <p>Anticipar, sin necesidad de ejecutarlo, qué modificaciones del escenario produce (qué hace) un programa en un entorno de enseñanza para la programación.</p>
--	--	--

Contenidos → B3.3. **Herramientas de lenguaje de programación.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Crear programas en entornos de enseñanza de programación por bloques icónicos que combinen comandos para resolver problemas: utilizar comandos primitivos, secuencias y repetición simple en la construcción de un programa.</p>	<p>Crear programas en entornos de enseñanza de programación por bloques que combinen comandos, expresiones y eventos para resolver problemas:</p> <ul style="list-style-type: none"> • Utilizar comandos primitivos, secuencias, alternativas condicionales y repeticiones simples en la construcción de programas. 	<p>Crear programas en entornos de enseñanza de programación por bloques que combinen comandos, expresiones⁴⁴ y eventos⁴⁵ para resolver problemas:</p> <ul style="list-style-type: none"> • Utilizar comandos primitivos, secuencias, alternativas condicionales⁴⁶, repeticiones simples⁴⁷ y procedimientos en la construcción de programas.

⁴⁴ Una expresión es una herramienta de un lenguaje de programación que permite describir información.

⁴⁵ Un evento en un lenguaje de programación es un mecanismo para disparar la ejecución de cierta parte del programa basándose en la ocurrencia de cierta situación en el entorno de trabajo. Ejemplos de situaciones que disparan un evento son el clic de un *mouse*, el apretar una tecla, girar el celular, el paso del tiempo , etc.

⁴⁶ Una alternativa condicional es una forma de combinar comandos o expresiones en un lenguaje de programación que permite elegir entre dos comandos o expresiones cuál se ejecuta según corresponda, basándose en una condición. Por ejemplo, basándose en lo informado por un sensor que indique si hay un obstáculo delante, se podrá pautar la ejecución de ir hacia adelante o rodear el obstáculo.

⁴⁷ Una repetición es una forma de combinar comandos en un lenguaje de programación que permite ejecutar la misma secuencia de comandos más de una vez. Existen formas de repetición simple, basadas en una cantidad fija de repeticiones, o condicional, basadas en una condición que debe cumplirse para terminar de repetir.

	<ul style="list-style-type: none"> • Utilizar expresiones sin operadores para construir condiciones que no necesiten de comparaciones (por ejemplo, el bloque provisto por el entorno que detecta si un personaje está tocando el borde). • Utilizar eventos de interacción con el usuario (por ejemplo, para responder al uso de teclas o del mouse) para definir programas interactivos. 	<ul style="list-style-type: none"> • Utilizar expresiones literales⁴⁸ (números y cadenas de caracteres⁴⁹ o <i>strings</i>), sensores, listas y operadores simples⁵⁰ en la construcción de programas. • Utilizar eventos de interacción con el usuario (por ejemplo, para responder al uso de teclas o del <i>mouse</i>) para definir programas interactivos. <p>Identificar las herramientas de lenguajes de programación aprendidas en los diferentes entornos con los que trabajan, reconociendo que estas son independientes del lenguaje específico utilizado o del “estado del arte” de la tecnología.</p> <p>Definir y utilizar procedimientos sin parámetros⁵¹ para plasmar estrategias de solución en el programa expresando subtareas.</p> <p>Utilizar expresiones y comandos de variables para conservar, acceder y modificar información (por ejemplo, el puntaje actual de un juego) durante la ejecución de un programa implementados de alguna manera (por ejemplo, con datos en el entorno y expresiones primitivas para acceder y comandos primitivos para modificar).</p> <p>Utilizar expresiones y comandos de listas para agregar, acceder y eliminar elementos de un conjunto.</p> <p>Reconocer que los datos involucrados en un programa pueden ser de distinto tipo (por ejemplo, números, letras, colores, cadenas de caracteres, etc.) y que el</p>
--	---	--

⁴⁸ Una expresión literal es una expresión que menciona literalmente al valor que describe. También se suele hablar de “un literal”. Por ejemplo, los literales numéricos son 0, 1, 2, 3, etc. En contraposición, el sensor kmALaCiudad también es una expresión numérica, pero no literal, porque no se sabe su valor hasta no estar ejecutándose el programa.

⁴⁹ El tipo cadena de caracteres (en inglés, *string*) es un tipo de datos que representa secuencias de letras, y que se pueden utilizar para representar palabras, oraciones o secuencias arbitrarias de caracteres sin restricciones.

⁵⁰ Un operador es una expresión que describe la transformación de uno o más datos dados como argumentos en una información resultado. Los operadores simples son operadores básicos, como la suma, la multiplicación, la negación, etc.

⁵¹ Un procedimiento es una herramienta de lenguajes de programación que permite definir un nuevo comando en el lenguaje.

		<p>tipo de dato está relacionado con cómo se puede operar con ellos (por ejemplo, los números se pueden sumar, las palabras se pueden concatenar).</p> <p>Considerar los tipos de datos involucrados en los argumentos de un comando para interpretar y solucionar errores de ejecución.</p> <p>Reconocer la importancia de construir programas que sean legibles (es decir, fácilmente entendibles por las personas) y utilizar las herramientas brindadas por el lenguaje de programación para conseguirlo (en particular, uso de procedimientos para la división en subtareas y denominación representativa de los identificadores).</p>
--	--	--

Contenidos → B3.4. **Ejecución secuencial y paralela de programas.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Comprender que al ejecutar el programa se llevan a cabo las acciones indicadas por los bloques en el orden definido en el programa.</p>	<p>Comprender que ejecutar un programa secuencial consiste en que el dispositivo o el artefacto computacional realice paso a paso las instrucciones especificadas en el programa.</p>	<p>Reconocer que existen modelos de ejecución secuencial, interactiva y de varios procesos secuenciales que se ejecutan en forma simultánea.</p>

Contenidos → B3.5. **Compilación e interpretación: intermediación entre el código fuente y la ejecución.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Reconocer la interpretación de un programa como proceso en el cual un programa (el intérprete) toma cada una de las instrucciones y las lleva a cabo efectivamente.</p> <p>Reconocer los entornos de programación por bloques como programas que interpretan los programas creados por las y los usuarios de dicho entorno.</p>

Área → C. Infraestructura tecnológica

Eje → C1. Organización y arquitectura de computadoras

Saberes ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

El reconocimiento de las computadoras como **máquinas programables** que **reciben, procesan y generan información** a partir de programas que se ejecutan sobre **componentes de hardware** como el procesador, la memoria y los dispositivos de entrada y salida.

La conceptualización de las computadoras como máquinas programables organizadas como sistemas en los cuales interactúan **componentes físicos (hardware)** y **programas (software)** para lograr el funcionamiento esperado.

La identificación de las funciones del procesador, la memoria RAM y los dispositivos de entrada y salida para comprender que todo sistema con este tipo de componentes es una forma de computadora, más allá de su forma física.

Contenidos → C1.1. **La computadora como un sistema integrado por hardware y software.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer la existencia de programas o aplicaciones de usuario en el uso de una computadora.</p> <p>Identificar que las computadoras pueden tener aspectos muy variados.</p> <p>Identificar junto con la guía del docente funcionalidades comunes con las que cuentan diferentes aplicaciones o programas que utilizan (símbolos, iconos, menú para acceder o guardar archivos, etc).</p>	<p>Comprender que los programas o aplicaciones con los que cuentan las computadoras (<i>software</i>) requieren de componentes físicos (<i>hardware</i>) para funcionar.</p> <p>Reconocer que existen diferentes aplicaciones o programas que pueden tener el mismo objetivo o funcionalidad.</p>	<p>Reconocer que una computadora es un sistema en el cual interactúan diferentes componentes de hardware y software que se comunican entre sí para que funcione de la forma esperada.</p> <p>Reconocer que tanto el <i>hardware</i> como el <i>software</i> son imprescindibles para el funcionamiento de una computadora.</p>

Contenidos → C1.2. **Modelo de máquina programable: arquitectura von Neumann.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Identificar que el comportamiento de una computadora es el resultado de ejecutar programas (por ejemplo, cuando programan un robot).</p>	<p>Identificar que una computadora es una máquina programable que ejecuta los programas creados por personas.</p>	<p>Reconocer que las computadoras no solo se encuentran en las computadoras de escritorio o portátiles, sino también en dispositivos que forman parte de la vida contemporánea (por ejemplo, en</p>

	<p>Reconocer que una computadora recibe, procesa y genera información.</p>	<p>teléfonos celulares, televisores, automóviles, lector de tarjeta de transporte, cajero automático, etc.).</p> <p>Comprender que una computadora es una máquina programable que recibe información, la procesa (realiza operaciones, comparaciones o modificaciones) o la almacena y genera nueva información (una acción, una imagen, un texto) sobre la base de un programa.</p> <p>Integrar las funciones de los componentes de hardware y software y la noción de la computadora como sistema con el modelo de máquina programable.</p>
--	---	--

Contenidos → C1.3. Componentes de *hardware*. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer dispositivos de entrada y salida (cámara, parlantes, grabadora, teclado, etc.) en situaciones de uso de computadoras.</p>	<p>Reconocer que el procesador y la memoria son componentes que están presentes en todas las computadoras y son necesarios para ejecutar los programas.</p>	<p>Conocer la función de los componentes fundamentales de una computadora: procesador, memoria RAM, dispositivos de almacenamiento y dispositivos de entrada y salida.</p> <p>Comprender que todos los datos digitales que conservamos están almacenados en algún dispositivo de almacenamiento (siempre existe un soporte físico, ya sea local o remoto).</p> <p>Conocer la existencia de diferentes unidades de medida para describir la capacidad de almacenamiento (KB, MB, GB).</p>

Contenidos → C1.4. **Ejecución a bajo nivel: lenguaje de máquina, representación binaria e impulsos eléctricos.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Eje → C2. **Sistemas Operativos** 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>El reconocimiento de la necesidad de un sistema operativo para poder instalar, abrir y cerrar otros programas o aplicaciones.</p> <p>La vinculación entre sus producciones digitales, los archivos que las guardan y los dispositivos en los que se almacenan.</p>		<p>El reconocimiento del sistema operativo como el programa central presente en los dispositivos computacionales, encargado de organizar el funcionamiento y la interacción entre <i>hardware</i> y <i>software</i>.</p> <p>El reconocimiento de los archivos como la unidad de almacenamiento de información y los directorios como una forma eficaz de organizarlos.</p>

Contenidos → C2.1. **El sistema operativo como administrador de recursos e interfaz para el *hardware*.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer que cuando una computadora se inicia, hay un programa especial (sistema operativo) que se ejecuta, y nos permite usar otros programas y aplicaciones.</p>	<p>Reconocer que en las computadoras hay una pieza clave del sistema que es un programa (el sistema operativo) que necesitamos para instalar, abrir y cerrar otros programas o aplicaciones.</p>	<p>Reconocer que los dispositivos computacionales que permiten usar varias aplicaciones (notebook, celular, PC, smart TV) tienen un sistema operativo (SO).</p> <p>Reconocer al SO como un programa central que organiza los componentes de <i>hardware</i> y <i>software</i> y al que recurren los programas para interactuar con el hardware.</p> <p>Conocer que existen diferentes sistemas operativos (Unix, Windows, Android OS, IOS, etc.).</p> <p>Reconocer que los programas que utilizamos habitualmente están contruidos para un sistema operativo en particular, y por ello es necesario atender con qué sistema operativo son compatibles los programas que queremos usar.</p> <p>Reconocer que para que el sistema operativo pueda controlar los componentes de <i>hardware</i> existe un <i>software</i> llamado controlador (<i>driver</i> en inglés) que se debe instalar para cada dispositivo y debe ser compatible con el sistema operativo.</p>

Contenidos → C2.2. **Abstracciones del sistema Operativo: archivos y sistemas de archivos, procesos, virtualización, usuarios.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer que sus creaciones digitales pueden ser guardadas en las computadoras que utilizan para ser recuperadas.</p>	<p>Asociar que al guardar nuestras producciones digitales, creamos un archivo que se almacena en un dispositivo de almacenamiento.</p>	<p>Reconocer que toda la información que conserva un dispositivo computacional se guarda en un archivo y todos los archivos se organizan en directorios y esto está reflejado en la ruta del archivo.</p> <p>Reconocer que para poder explorar, recuperar, modificar y eliminar información de un dispositivo computacional es importante organizar de forma eficaz los directorios y utilizar nombres para los archivos y directorios que se relacionen con el contenido o la información que almacenan.</p> <p>Reconocer que los archivos cuentan con propiedades: nombre, tipo, tamaño, autor, permisos (escritura, lectura, ejecución), fechas de creación y modificación.</p> <p>Reconocer que la extensión de archivo es parte del nombre, que nos permite declarar el tipo del archivo, y que es información que el sistema operativo puede utilizar para elegir un programa para abrirlo.</p> <p>Reconocer que el SO brinda herramientas al usuario administrador para generar otros usuarios con diferentes permisos y limitaciones de acceso a archivos y programas.</p>

Eje → C3. Redes e internet 

Saberes ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

El reconocimiento de **internet** como una **red de computadoras** que **intercambian información** entre sí, algunos de sus **componentes físicos** fundamentales y una **aproximación al funcionamiento de las aplicaciones** en línea.

La comprensión de **internet** como una **red de computadoras** que intercambian información entre sí, sus componentes físicos fundamentales, su organización y el **recorrido de la información**, para reconocer la necesidad de **mecanismos de seguridad** para un intercambio seguro y de un **servidor** que permita a las aplicaciones de internet acceder, almacenar y procesar la información.

Contenidos → C3.1. Estructura y funcionamiento de redes de computadoras. Internet como una red. 

Alcances ↓

Inicial

Primaria, primer ciclo

Primaria, segundo ciclo

Comprender a internet como el **canal** a través del cual las computadoras se comunican entre sí.

Reconocer que el **wifi** es la forma en la que conectamos de forma inalámbrica una computadora a internet.

Reconocer que internet es una **red global** de computadoras que permite a las personas conectarse con otras en todo el mundo a través de muchas computadoras que funcionan como **puntos de conexión intermedios**.

Identificar algunos **dispositivos físicos necesarios** para conectarnos a internet (por ejemplo, un router wifi).

Conocer las **redes** de computadoras como un conjunto de **computadoras interconectadas** entre sí de manera que pueden **intercambiar información** entre ellas.

Reconocer distintas redes de computadoras que forman parte de la **vida contemporánea** (por ejemplo, redes domésticas o escolares), algunos de sus

		<p>componentes fundamentales (routers y hosts), su organización y el recorrido de la información.</p> <p>Identificar internet como la red de computadoras que se comunican entre sí intercambiando información a escala mundial y con millones de usuarios.</p> <p>Reconocer los enlaces que forman una red y las tecnologías más usuales (Ethernet, wifi, 4g).</p> <p>Reconocer los enlaces físicos sobre los que se construye internet a nivel mundial, como los cables de enlace submarino, enlaces satelitales, etc.</p> <p>Reconocer que el envío de información a través de una red requiere del paso por puntos intermedios.</p> <p>Conocer que al enviar información a través de internet, se traslada por medios físicos, se fragmenta en paquetes para su traslado, se recompone al llegar a destino y que cada paquete puede seguir una ruta diferente.</p> <p>Identificar si la imposibilidad de conectarse a internet corresponde a un problema del servicio o algún problema de los dispositivos de la red local.</p>
--	--	---

Contenidos → C3.2. **Aplicaciones en internet: almacenamiento remoto, www, modelo cliente-servidor y P2P.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Reconocer que algunos de los programas o aplicaciones que usan necesitan que las computadoras estén conectadas a internet.</p>	<p>Identificar en el funcionamiento de algunas aplicaciones en línea, la existencia de una computadora conectada</p>	<p>Comprender que las aplicaciones que funcionan en internet están conformadas por diversos programas</p>

a internet que almacena la información que necesitamos (por ejemplo, un video en línea).

que funcionan en diferentes computadoras intercambiando información a través de la red.

Reconocer que el funcionamiento de ciertas aplicaciones en red (en particular internet) necesitan de un **servidor**, es decir, de una computadora que esté disponible todo el tiempo, para poder acceder, almacenar y procesar información.

Distinguir **qué información reside en su dispositivo** de cómputo (computadora, celular, consola, etc.) y cuál en servidores o en otros dispositivos.

Reflexionar sobre la metáfora de “**la nube**” y su concreción a través de servidores que se encuentran en grandes centros de datos.

Reconocer que la información y los datos que se suben o comparten en internet (redes sociales, aplicaciones, videojuegos online, etc.) son almacenados en **servidores controlados por las empresas dueñas de los servicios utilizados.**

Reconocer distintas aplicaciones (un navegador, un cliente de mensajería, una aplicación de videoconferencia, una aplicación de contenidos multimedia) que **utilizan internet como un canal de comunicación.**

Reconocer los navegadores, repositorios remotos de almacenamiento, servicios web, entre otros, como distintas categorías de las funciones de las aplicaciones de internet (Firefox, Chrome, Chromium, Opera, Safari cumplen la función de navegación).

Identificar la **web** (World Wide Web o simplificado www) como un sistema particular en el que un navegador (Firefox, Chromium, etc.) intercambia información a través de internet con un servidor y la visualiza en formato de páginas web.

Contenidos → C3.3. **Mecanismos de seguridad en redes: protocolos, criptografía y anonimización.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Comprender que el uso de redes expone al usuario a diferentes ataques o problemas de seguridad porque la computadora puede ser accesible desde cualquier otra computadora de la red y la información enviada puede ser interceptada por las computadoras intermedias.</p> <p>Conocer la necesidad de la encriptación como forma de proveer seguridad en el intercambio de información en redes y reconocer que está presente en muchas aplicaciones.</p> <p>Conocer que a los usuarios de una red se le pueden asignar diferentes permisos (por ejemplo, solo acceder a internet y no al resto de los dispositivos de la red).</p>

Área → D. Datos e Inteligencia artificial 

Eje → D1. Recolección y modelado de datos 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	El reconocimiento y la experiencia de situaciones que involucran recolección de datos en forma manual y automatizada.

Contenidos → D1.1. Recolección de datos activa y pasiva. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Reconocer diferentes formas de recolectar datos, que pueden o no involucrar dispositivos computacionales.</p> <p>Conocer herramientas automatizadas de recolección de datos como encuestas en línea.</p> <p>Reconocer en el funcionamiento de los artefactos computacionales contemporáneos la recolección de datos, en qué forma sucede y para qué son utilizados.</p>

Contenidos → D1.2. **Modelado de datos: entidades con atributos. Base de datos.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	Identificar en un registro de datos (por ejemplo los resultados de una encuesta en una planilla) cómo se registra cada ítem (por ejemplo, las respuestas de cada persona en una fila) y la información asociada a ese ítem (por ejemplo, la respuesta a cada pregunta en una columna).

Eje → D2. **Ciencia de datos, inteligencia artificial y aprendizaje automático** 

Saberes ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
El conocimiento de que el comportamiento de algunas aplicaciones basadas en inteligencia artificial (recomendadores, clasificadores) reproducen ejemplos que les dieron previamente las personas .		El reconocimiento de la utilización de los datos para sacar conclusiones (a través de su visualización) y para entrenar modelos de aprendizaje automático que resuelven problemas de clasificación.

Contenidos → D2.1. Aplicaciones de inteligencia artificial. Mecanismos generales. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
<p>Identificar que las recomendaciones de videos, películas o canciones que realizan algunas aplicaciones dependen de lo que vimos o escuchamos anteriormente en esa misma aplicación.</p>	<p>Reconocer la forma en que puede crearse una aplicación de inteligencia artificial (IA) para identificar la existencia de un conjunto de datos de entrenamiento, el rol de las personas en su clasificación y que lo que hacen las aplicaciones es intentar reproducir esa clasificación.</p> <p>Comparar cómo generan respuestas las aplicaciones de IA con cómo lo hacemos las personas y qué tipo de errores comete cada una.</p>	<p>Conocer que la mayoría de las aplicaciones de IA contemporáneas dependen de enormes volúmenes de datos y están construidas con técnicas de aprendizaje automático (AA).</p> <p>Comprender que el entrenamiento de un modelo o aprendizaje de máquina consiste en encontrar patrones de forma automática y que, por lo tanto, no constituye una forma de aprendizaje como lo hacemos los humanos, basada en el razonamiento.</p>

Contenidos → D2.2. Preparación, análisis y visualización de datos. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Identificar que la visualización de datos es una herramienta tanto para la comunicación como para el análisis de datos.</p> <p>Crear gráficos estándares (de barra, de tortas, de líneas, etc.) para comunicar o analizar frecuencias de</p>

		<p>elementos de un conjunto de datos utilizando las herramientas de las planillas de cálculo.</p> <p>Reconocer que un mismo conjunto de datos puede ser presentado de diferentes formas para resaltar distintos aspectos (por ejemplo, una tabla de resultados se puede ordenar según la cantidad de derrotas, victorias o puntos, dependiendo de lo que se considere relevante).</p>
--	--	--

Contenidos → D2.3. Modelos de aprendizaje automático. 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	<p>Reconocer aplicaciones contemporáneas cuyo funcionamiento depende de un conjunto de datos y, por lo tanto, pueden estar implementadas con aprendizaje automático (AA).</p> <p>Crear modelos de aprendizaje automático de clasificación utilizando entornos diseñados para la enseñanza de IA:</p> <ul style="list-style-type: none"> • Analizar el problema para definir las clases. • Seleccionar y clasificar conjuntos de datos (imágenes, sonidos, etc). • Elegir datos para evaluar si el modelo responde a lo esperado. <p>Reconocer que las aplicaciones basadas en AA funcionan a partir de reconocer y replicar patrones detectados automáticamente en enormes volúmenes de datos.</p>

		Reconocer que el desarrollo de aplicaciones con aprendizaje automático a nivel industrial necesita enormes volúmenes de datos y de muchísimo poder de cómputo .
--	--	---

Contenidos → D2.4. **Subjetividad expresada en los datos. Sesgo algorítmico.** 

Alcances ↓

Inicial	Primaria, primer ciclo	Primaria, segundo ciclo
--	--	--

Saberes y alcances por nivel educativo

Nivel Secundario

Área → **A. Ciudadanía y Computación** 

Eje → **A1. Identidad digital** 

Saberes ↓

Secundaria, ciclo básico

El reconocimiento de **los intereses que guían el desarrollo de artefactos y dispositivos computacionales**, las consecuencias y desigualdades para la ciudadanía.

Secundaria, ciclo orientado

La **actitud crítica sobre los artefactos y dispositivos computacionales** en términos de poder y concentración, las consecuencias para la ciudadanía y la soberanía, y la necesidad de contar con un marco regulatorio que proteja los derechos de la ciudadanía.

Contenidos → **A1.1. Convivencia en internet. Identidades digitales.** 

Alcances ↓

Secundaria, ciclo básico

Comprender que los entornos digitales son **espacios de convivencia** donde se ejerce la **ciudadanía**, reconociendo derechos y responsabilidades propios y respetando y reconociendo los derechos y responsabilidades de las otras personas.

Conocer que en la Argentina existe una ley (Ley 27078, Tecnologías de la información y las comunicaciones) que establece **derechos digitales sobre el acceso a la tecnología**.

- Todas las personas tienen derecho a usar computadoras, celulares y dispositivos electrónicos.

Secundaria, ciclo orientado

Ser partícipes activos de espacios de convivencia en internet respetuosos.

Comprender la importancia de contar con y defender los **derechos digitales de la ciudadanía**.

- Tienen derecho a tener internet sin importar edad o género, niveles de ingreso, ubicación geográfica o si tienen alguna discapacidad.

Ampliar el **horizonte cultural** reconociendo los espacios digitales como ámbitos de participación y compromiso donde ejercer una ciudadanía activa.

Vincular las nociones de **estructura y funcionamiento** de la red internet con las posibilidades de circulación de información, el alcance y la inmediatez para comprender los diferentes **usos y prácticas sociales** que esta estructura habilita.

Identificar los **discursos de odio** que son favorecidos por algunas de las características de internet (como el anonimato o la falta de regulación y mediación de la información), para no apoyarlos ni promoverlos, así como también conocer las formas de denunciarlos⁵².

Analizar críticamente las **implicancias de la publicación de información** (datos personales, imágenes, videos, noticias, comentarios, gustos, amistades) **en la identidad digital propia y ajena** considerando qué se publica, dónde se publica y con qué grado de visibilidad, teniendo en cuenta que los entornos digitales son espacios de convivencia entre personas. Así como reconocer la importancia de pedir el **consentimiento** si la información involucra a otra persona.

Recuperar los conocimientos sobre **ciberbullying**, y vincularlos con el desarrollo de una identidad digital responsable sin perjudicar a terceros.

⁵² Se entiende por “discursos de odio” ciertas narrativas sociales que circulan y se reproducen principalmente en el espacio público, ya sea en los medios de comunicación, redes sociales e internet, y que transmiten prejuicios y estereotipos sobre un grupo o colectivo de personas en particular, teniendo por objetivo justificar, legitimar e incitar la confrontación y/o la violencia social de un sector de nuestra sociedad sobre otro. (INADI, [Una aproximación a los discursos de odio: antecedentes de investigación y debates teóricos. Coordinación de Investigaciones y Observatorios sobre Discriminación](#))

Contenidos → A1.2. Información pública y privada compartida en internet. 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Conocer que existe una **ley de protección de datos personales**⁵³ (Ley 25326) que protege a las ciudadanas y ciudadanos si sus datos de identidad, salud o crédito son usados sin su consentimiento, y que garantiza que puedan saber quién, para qué, cuándo y por qué se utilizan sus datos personales guardados en archivos, registros, bancos de datos públicos o privados.

Saber que nadie puede obligar a una persona a compartir **datos sensibles** que revelen origen racial y étnico, opiniones políticas, convicciones religiosas, filosóficas o morales, afiliación sindical e información referente a la salud o a la vida sexual, y que es una decisión personal **hacerlos públicos** en función de cómo cada quien quiera **construir su identidad digital**.

Comprender por qué los datos que puedan ser usados para **identificar, contactar o localizar** a una persona, datos **bancarios y de crédito** son también datos sensibles, y formar criterios para determinar cuándo las acciones personales **exponen** estos datos propios o de terceros pudiendo comprometer la **seguridad** o **privacidad** de una persona.

Gestionar la **privacidad** identificando ventajas, desventajas, riesgos y posibilidades al compartir **información como pública y como privada** en entornos digitales para decidir qué y cómo publicarla.

--

⁵³ <https://www.argentina.gob.ar/justicia/derechofacil/leysimple/datos-personales>

Contenidos → A1.3. **Huella Digital.** 

Alcances ↓

Secundaria, ciclo básico

Conceptualizar el fenómeno de la huella digital como todos los **datos que resultan de nuestra actividad en internet**, teniendo en cuenta que los datos que generamos para que sean registrados (un “me gusta”, comentarios, suscripciones, etc.) y los que resultan del registro automático y continuo de nuestras acciones en internet (clics, sitios visitados, búsquedas, tiempo de permanencia, ubicación geográfica, etc.) quedan registrados y eventualmente pueden ser utilizados.

Formar **criterio para decidir** si se deja información de forma voluntaria, considerando que deja un **rastro** que eventualmente **pueda ser usado** para fines como publicidad dirigida, campañas políticas personalizadas, recomendaciones de contenido, predicciones, etc.

Secundaria, ciclo orientado

Reflexionar acerca de la huella digital como un **rastro económicamente valioso**.

Contenidos → A1.4 **Vulneración de derechos y delitos informáticos. Grooming.** 

Alcances ↓

Secundaria, ciclo básico

Identificar las formas particulares que adquiere la **violencia de género en entornos digitales** (la reproducción de estereotipos; los discursos de odio; recibir contenidos sexuales explícitos sin consentimiento; exigir el envío de imágenes o información íntima; difusión de imágenes íntimas sin consentimiento; sextorsión; ejercer la

Secundaria, ciclo orientado

--

vigilancia digital exigiendo compartir ubicación, compartir contraseñas, mostrar contactos o conversaciones, etc.) y conocer las **formas de buscar ayuda**.

Conceptualizar el **grooming** como toda acción por la que una persona adulta contacta a una niña, un niño o adolescente a través de entornos digitales para atentar contra su integridad sexual, saber que se encuentra penado (Ley 27590⁵⁴); adquirir habilidades para reconocer estas situaciones y saber qué hacer ante un caso de grooming, así como conocer sobre la existencia de canales estatales de ayuda.

Reconocer **modalidades de delitos informáticos**⁵⁵ contra la libertad, la propiedad (estafas), entre otros, para cuidarse, buscar ayuda y vincularlo a las nociones de infraestructura y mecanismos de seguridad en redes (por ejemplo, tener los cuidados necesarios al usar una red wifi de acceso público).

Eje → A2. Estrategias para un uso seguro de internet

Saberes ↓

Secundaria, ciclo básico

La anticipación, descripción y desarrollo de **pautas de seguridad** para prevenir situaciones de acoso, delitos informáticos y robo de identidad en internet.

Secundaria, ciclo orientado

La **anticipación, descripción y desarrollo colectivo de pautas de seguridad** para usuarios, a través de la elaboración y apropiación de formas de proceder para prevenir y protegerse de ataques y acoso cibernéticos (por ejemplo, *phishing*, *malware* y *ransomware*, *ciberbullying*, etc.) y del uso responsable de información privada publicada en aplicaciones en internet (tales como redes sociales, videojuegos, servicios de mensajería, etc.).

⁵⁴ <https://www.argentina.gob.ar/desarrollosocial/grooming>

⁵⁵ <https://www.argentina.gob.ar/justicia/derechofacil/leysimple/delitos-informaticos>

Contenidos → A2.1 **Estrategias para la protección de la información privada y la identidad digital: claves seguras, permisos y sitios seguros.** 

Alcances ↓

Secundaria, ciclo básico

Afianzar la práctica de generación y utilización de **claves de acceso** (contraseñas) y los criterios de seguridad asociados a estas para diferenciar claves seguras y no seguras.

Reconocer **situaciones que involucran contraseñas y que suponen un riesgo para la seguridad** y saber cómo manejarlas (por ejemplo, no usar la contraseña en una red pública, no usar la misma contraseña para diferentes aplicaciones, modificar la clave de forma periódica).

Conocer los **tipos de permisos** disponibles en diferentes aplicaciones de internet, y cómo la administración de esos permisos permite controlar los **grados de visibilidad**, protegiendo la privacidad y la identidad digital.

Adquirir criterios para evaluar las **garantías de seguridad** (acceso con clave, autenticación, comunicación con protocolos seguros y solicitud solo de los permisos necesarios) que ofrece un sitio o aplicación en relación con la información personal que solicita (por ejemplo, proveer datos de tarjeta de crédito en sitios seguros, no brindar permisos de acceso a contactos a una aplicación de linterna).

Considerar estrategias diversas⁵⁶ para **proteger los datos personales**.

Considerar los cuidados necesarios para **sextear**⁵⁷ **de forma segura** sin que la información intercambiada se vuelva pública.

Secundaria, ciclo orientado

Apropiarse y generalizar estrategias para la protección de la información privada y la identidad digital.

⁵⁶ Algunas estrategias pueden ser: usar el modo incógnito para que no se guarden las contraseñas y el historial de navegación, no usar la misma contraseña para los sitios a los que se accede y para las redes sociales, no ingresar datos personales en sitios desconocidos, no responder correos electrónicos donde solicitan que se completen datos personales, leer los términos y condiciones de uso de los datos personales que proponen las redes sociales o aplicaciones antes de aceptarlas, no guardar contraseñas en lugares públicos, en sitios web que requieren el ingreso de usuario y contraseña, verificar que la dirección de la página sea auténtica. <https://www.argentina.gob.ar/justicia/convosenlaweb/situaciones/como-cuidar-los-datos-en-la-nube>

⁵⁷ El sexting forma parte de las maneras de expresar la sexualidad, en este caso particularmente a través de medios digitales <https://www.argentina.gob.ar/justicia/convosenlaweb/situaciones/como-hago-para-sextear-seguro>

Contenidos → A2.2 Estrategias para la protección contra *software* malicioso, rastreadores, sitios falsos y *phishing*. 

Alcances ↓

Secundaria, ciclo básico

Reconocer la existencia de **mecanismos para recopilar información de navegación** en internet (*cookies*, rastreadores), sitios o aplicaciones diseñadas con propósitos dañinos (*software* malicioso o *malware*) y complementos para el navegador que los identifican o bloquean.

Identificar **mensajes fraudulentos** que contienen enlaces donde se solicitan datos privados sensibles (por ejemplo, credenciales de acceso o números de tarjeta de crédito) o se induce a **acciones perjudiciales** (como el envío de dinero a cuentas ajenas), práctica que se conoce como *phishing*.

Comprender que el usuario es un **participante activo** de los ataques producidos por *malwares*, uso de sitios inseguros y *phishing*.

Secundaria, ciclo orientado

Apropiarse y generalizar **estrategias para la protección** contra *malware*, sitios falsos y *phishing*.

Apropiarse y generalizar estrategias para la protección de la huella digital.

Eje → A3. Computación y sociedad 

Saberes ↓

Secundaria, ciclo básico

El reconocimiento de los **intereses** que guían el desarrollo de artefactos y dispositivos computacionales, las consecuencias y desigualdades para la ciudadanía.

Secundaria, ciclo orientado

La **actitud crítica** sobre los artefactos y dispositivos computacionales en términos de poder y concentración, las consecuencias para la ciudadanía y la soberanía, y la necesidad de contar con un marco regulatorio que proteja los derechos de la ciudadanía.

Contenidos → A3.1. **Desarrollos computacionales: evolución, oportunidades y brechas, intereses y necesidades. Su influencia en la vida y los derechos de las personas. Soberanía tecnológica.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer que los desarrollos computacionales influyen en las **costumbres** de las personas y en sus **consumos culturales**, y crean **nuevas necesidades**.

Reflexionar sobre el hecho de que el desarrollo de dispositivos y artefactos computacionales es motivado por **intereses y necesidades** de personas, organizaciones o empresas.

Reconocer la **influencia desigual en los desarrollos computacionales de los intereses económicos o políticos** de grupos reducidos de personas, organizaciones o empresas por sobre los de la población.

Comprender desde una **perspectiva histórica, política y de género** el desarrollo de la computación.

Reconocer que la **desigualdad de género** adquiere nuevas formas en internet:

- las mujeres y personas del colectivo LGTBIQ+ tienen menos posibilidades de acceder, saber y usar (brecha digital), tienen menos representación en la industria del *software* y, por tanto, los desarrollos no responden a diversos intereses y necesidades.
- Los contenidos digitales que circulan de forma masiva reproducen estereotipos de género.

Secundaria, ciclo orientado

Reflexionar sobre cómo la irrupción de nuevos artefactos tecnológicos y computacionales influyeron e influyen en la **modificación de actividades laborales y profesionales**, y reflexionar sobre las consecuencias de ello en la vida de las personas.

Reflexionar acerca de las implicaciones sociales, políticas y económicas de la **concentración de servicios y productos tecnológicos en pocas empresas** (por ejemplo, con respecto a los derechos de las personas).

Reconocer el desigual desarrollo computacional en diferentes partes del mundo reflexionando particularmente sobre el **desarrollo local en contraste con el mundial**, y las consecuencias de este contraste con el desarrollo de la **soberanía tecnológica** a nivel nacional.

Reflexionar sobre cómo una solución tecnológica o computacional generada en una región e impulsada de modo global puede **impactar de modo diferente en culturas o sociedades y generar nuevos problemas**.

Comprender la importancia de contar con un marco regulatorio y exigir que se aplique, para proteger los **derechos de las personas ante las consecuencias de desarrollos computacionales**.

Ser partícipes activos y agentes de cambio en la promoción de la equidad de género vinculada a los desarrollos computacionales.

Contenidos → A3.2. **Sesgos, prejuicios y estereotipos. Filtros en el acceso a la información en internet.** 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Reconocer que existen **criterios e intereses** según los cuales los artefactos computacionales realizan el **recorte en la información** a la que accedemos en internet (personalizaciones, legislación, intereses de los propietarios), conocido como **burbuja de filtros** ⁵⁸, y que puede generar censura u homogeneización de la información.

Reconocer de qué manera los artefactos computacionales están atravesados por las **subjetividades y los intereses** de quienes los desarrollan⁵⁹.

Reconocer situaciones en las que el sesgo algorítmico **reproduce o refuerza prejuicios o discriminaciones** y dimensionar sus **consecuencias en términos humanos** , por ejemplo, la censura, el aislamiento, la amplificación de estereotipos, etc.

Tomar conciencia de que la percepción que se construye de la realidad a partir de la información a la que accedemos en **internet** es **parcial y sesgada** .

Comprender la necesidad de contar con un marco regulatorio y exigir que se aplique, para proteger los **derechos de las personas** ante las consecuencias del desarrollo y aplicación de los **artefactos computacionales basados en datos** (por ejemplo, para qué tipos de problemas se pueden aplicar modelos de inteligencia artificial).

--

⁵⁸ El concepto de burbuja de filtros se refiere a la selección personalizada de la información que recibe cada individuo al consumir contenido en internet. Esta lo hace sentir cómodo, pero lo aísla de las y los demás. Los algoritmos que utilizan las redes permiten recopilar todos los rastros de información digital que va dejando cada cibernauta —desde búsquedas en Google a “Me gusta” en Facebook—, y van aprendiendo sobre esa persona para ofrecerle una visión del mundo ajustada a sus preferencias. De esta forma, un mismo término introducido por dos personas distintas en un motor de búsqueda no arrojará ni la misma cantidad ni los mismos resultados para una y para otra, sino que los adaptará en función de lo que conoce de cada una, gustos personales, creencias, orientación ideológica. (Véase [la conferencia de Eli Pariser en Ciclo #TechSociety, 2017](#), Fundación Telefónica).

⁵⁹ Por ejemplo, una herramienta de recolección que no contempla entre sus opciones diversas opciones de género.

Contenidos → A3.3. **El problema del poder, el control y la vigilancia en las redes. Neutralidad de la red.** 

Alcances ↓

Secundaria, ciclo básico

Identificar cómo la estructura de internet hace posible formas de **vigilancia** sobre el comportamiento de las personas (por ejemplo, a través del análisis del tráfico de datos) y cómo el control de tráfico favorece o limita la **neutralidad de la red**⁶⁰.

Secundaria, ciclo orientado

Conocer y adquirir conciencia sobre la responsabilidad que conlleva el **poder** que tienen los **proveedores de servicios de internet**, en tanto pueden favorecer o limitar la neutralidad de la red y ejercer formas de vigilancia y control.

Comprender la importancia de contar con un marco regulatorio y exigir que se aplique para controlar la **neutralidad de la red**, limitar la vigilancia y distribuir el control en función de los beneficios o derechos digitales de la ciudadanía.

Contenidos → A3.4. **Desinformación a través de internet.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer cómo influye en la población la propagación de **discursos de odio**, la generación y circulación masivas de **información falsa o engañosa**.

Secundaria, ciclo orientado

Reconocer que la circulación de la información falsa, engañosa o con discursos de odio pueden constituirse como **herramientas de poder**.

⁶⁰ “La **neutralidad de la red** fue concebida para proteger el carácter libre, abierto e innovador de internet, para lo cual se garantizan dos cosas: cualquiera debe poder conectarse y la información debe fluir lo más libremente posible [...] la neutralidad de la red establece que los operadores de redes (incluidos los proveedores de servicio de internet al hogar) no deben discriminar arbitrariamente entre la información que pasa por su infraestructura. Es decir, deben permitir todo uso legítimo de la red sin discriminar arbitrariamente por origen, destino, dispositivos, protocolos o puertos”. Cita de Javier Pallero (2018), “[Neutralidad de la red: ¿qué te puedo cobrar?](#)”, Buenos Aires: El Gato y la caja. La Ley argentina digital 27078 en sus artículos 56 y 57 contempla este derecho de la ciudadanía y regula a los proveedores de servicios de internet.

Conocer mecanismos para la **viralización** de información falsa (por ejemplo, bots y troll centers) y cómo los **algoritmos** contribuyen a amplificarla.

Reconocer la existencia de soluciones computacionales que facilitan la **generación y propagación de información falsa** (*deep fakes* a partir de montajes de video o sintetizadores de voz) y el trabajo de **análisis manual** para detectar si una información es falsa o engañosa.

Conocer e implementar estrategias para **detectar información falsa** (verificación de fuentes, distinguir si se refieren a hechos u opiniones, etc.).

Reconocer los **intereses y motivaciones** de las personas o instituciones detrás de la **manipulación de la información** (por ejemplo, noticias falsas con la intención de confundir, influenciar las opiniones e impactar en ciertos sucesos sociales como los procesos electorales de autoridades).

Incorporar estrategias y mecanismos para **identificar una noticia falsa** y saber cómo proceder para controlarla, denunciarla, etc.

Reconocer que la **desinformación** utiliza como vehículo los discursos de odio aprovechando que, al apelar a la emocionalidad, este tipo de mensajes promueven mayores reacciones en las personas, lo que ayuda a amplificar su difusión.

Comprender la importancia de contar con un marco regulatorio para denunciar y penalizar la información falsa, y exigir que se implemente.

Contenidos → A3.5. Modelos de producción y distribución de *hardware* y *software*.

Alcances ↓

Secundaria, ciclo básico

Identificar al *software* como una **producción intelectual** (de personas, organizaciones o empresas) y, por lo tanto, sujeta a las nociones de derecho de autoría.

Reconocer que quienes desarrollan *software* disponen de **instrumentos legales** para habilitar o restringir modos de producción y de distribución (por ejemplo, las licencias Creative Commons vs. las patentes), y conocer sus características.

Reconocer la relación entre las **motivaciones y los intereses** de productores y usuarios de *software* y *hardware* con los **modos de producirlos o distribuirlos**.

Reconstruir desde una **perspectiva historiográfica** cómo se ha producido y compartido el *software* y el *hardware* desde sus inicios hasta el desarrollo de los grandes monopolios actuales.

Secundaria, ciclo orientado

Vincular el desarrollo de **instrumentos legales** con la defensa de algunos de los **intereses** de productores o personas usuarias.

Valorar la **producción colaborativa de software** como una forma de garantizar las **libertades** (libertad de usar, de estudiar, distribuir y mejorar) y los **derechos** vinculados a los artefactos y dispositivos computacionales.

Apropiarse de **prácticas de participación computacional, cultura libre** y comunicabilidad de los programas desarrollando y utilizando **bibliotecas colaborativas de funciones y procedimientos** con archivos y programas legibles y comprensibles por la comunidad de esa aula.

Contenidos → **A3.6. Propiedad de los datos. Regulación, términos y condiciones.** 

Alcances ↓

Secundaria, ciclo básico

Identificar en qué medida la información y los datos que se suben o comparten en internet son **propiedad** de las empresas dueñas de los servicios y las posibilidades que esto les otorga (por ejemplo, la potestad de lucrar comercialmente con ellos o venderlos), recurriendo a los términos y condiciones.

Reconocer que la propiedad y uso de **datos a gran escala** puede ser una herramienta para **influir en fenómenos sociales** (elecciones, campañas publicitarias, etc.).

Tomar conciencia de que al aceptar los **términos y condiciones** de una plataforma se obtienen servicios **a cambio de algo** (por ejemplo, información personal o preferencias de uso o consumo del usuario).

Reflexionar sobre las consecuencias del uso que hacen las empresas u organismos públicos propietarios de los datos, vinculando con la necesidad de **instrumentos de regulación** (términos y condiciones de las plataformas, leyes de protección de datos personales, derecho al olvido, etc.).

Secundaria, ciclo orientado

--

Contenidos → A3.7. **Computación y ambiente: obsolescencia programada y derecho a reparar, extractivismo, demanda energética e impacto ambiental.** 

Alcances ↓

Secundaria, ciclo básico

Comprender quiénes se benefician y quiénes se perjudican con la **obsolescencia programada** y el **derecho a reparar**.

Tomar conciencia de la dimensión de la **demanda de infraestructura física y energética** de los servicios de **internet** (como redes sociales, buscadores, servicios de mensajería, video, etc.) y su **impacto ambiental** asociado a los grandes centros de cómputos.

Secundaria, ciclo orientado

Poner en relación el impacto de la producción y uso de dispositivos y artefactos computacionales con la noción de **sustentabilidad** definida por la ley de Educación Ambiental Integral.⁶¹

⁶¹ La sustentabilidad, en el marco de la Ley 27621 de educación ambiental, se define como un proyecto social que integra el desarrollo con justicia social, la distribución de la riqueza, la preservación de la naturaleza, la igualdad de género, la protección de la salud, la democracia participativa y el respeto por la diversidad cultural, y que busca el equilibrio entre diversas dimensiones, como la social, la ecológica, la política y la económica en el marco de una ética que promueve una nueva forma de habitar nuestra casa común. (Véase Ley de [Educación ambiental. en InfoLEG - Ministerio de Justicia y Derechos Humanos - Argentina](#)).

Área → B. Programación 

Eje → B1. Soluciones a problemas computacionales 

Saberes ↓

Secundaria, ciclo básico

La resolución de problemas computacionales a través del diseño de **estrategias** que permitan **estructurar y construir programas**, incorporando la dimensión de los datos (representación de datos y transformaciones sobre ellos), siempre en vinculación con las herramientas específicas de lenguajes de programación.

La comprensión de **características importantes en el proceso de programación**, como las capas de abstracción, la legibilidad, y otras que contribuyen a expresar mejor las soluciones.

Secundaria, ciclo orientado

El desarrollo de **criterios para evaluar e implementar soluciones** desde distintos puntos de vista (legibilidad, modularidad, reutilización y eficiencia) teniendo en cuenta las necesidades del problema.

La comprensión de nociones de **computabilidad y complejidad**, el reconocimiento de los **algoritmos y estructuras de datos** como soluciones computacionales generales.

Contenidos → B1.1. **Diseño de programas: estrategias de solución, modularidad y legibilidad.** 

Alcances ↓

Secundaria, ciclo básico

Adquirir fluidez en el uso de la técnica de división de subproblemas para elaborar **estrategias** de solución, y crear programas utilizando las herramientas del lenguaje disponibles.

Considerar los **datos** involucrados en un problema como parte de la estrategia de solución.

Identificar qué **datos** existen sólo **durante la ejecución de un programa** y cuales necesitan ser almacenados de **forma persistente**.

Reconocer las **interfaces de programación** como el mecanismo para acceder a un conjunto de procedimientos y funciones que definen **lo que se puede hacer y conocer** sobre una entidad o aspecto del problema sin necesariamente conocer las soluciones⁶².

Comprender las **interfaces de programación** como una herramienta que permite **reutilizar** procedimientos y funciones predefinidos y utilizarlos sin conocer cómo fueron desarrollados, pero comprendiendo su utilidad, por lo que constituyen una primera aproximación de la noción de **capa de abstracción**.

Entender la noción de **legibilidad de un programa** como la cualidad de un programa que permite que otra persona, al leer el código, pueda comprender la solución propuesta (por ejemplo, que el nombre elegido para una variable describa la intención del dato que debe guardarse allí).

Secundaria, ciclo orientado

Adquirir fluidez en el uso de la **técnica de división de subproblemas** para crear programas y definir bibliotecas⁶⁴ utilizando las herramientas del lenguaje disponibles.

Incorporar la noción de **modularización del código** al diseño de soluciones, a través de la definición de procedimientos y funciones y de bibliotecas de código.

Extender la noción de **legibilidad** al uso y definición de bibliotecas y tipos de datos estructurados.

Utilizar la modularización y legibilidad para diseñar soluciones **reutilizables**.

Reconocer la existencia de bibliotecas para decidir cuál es conveniente para el problema planteado.

Discernir en qué **nivel de abstracción** trabajar en función del problema a resolver (biblioteca para trabajar sobre videos, o utilizar una biblioteca que trabaje directamente sobre píxeles).

Estimar la **eficiencia** de un programa en términos de la cantidad de operaciones y de consumo de memoria.

Reconocer si una solución computacional es **implementable** teniendo en cuenta las limitaciones de eficiencia del dispositivo computacional.

⁶² Por ejemplo, los bloques que nos provee la extensión del lápiz en Scratch es su interfaz, y las **bibliotecas de funciones** poseen una interfaz que permite trabajar con ellas.

⁶⁴ En computación, una biblioteca (del inglés *library*, popularizado también como “librería”) es un módulo de *software* que ofrece una interfaz bien definida para interactuar con un conjunto de operaciones vinculadas temáticamente. A diferencia de un programa, no se espera que la biblioteca sea utilizada en forma autónoma, sino que su fin es ser utilizada en otros programas para resolver problemas específicos sin conocer los detalles de la implementación de las soluciones. A través del uso de bibliotecas bien definidas es posible construir programas modularmente de forma mucho más eficaz.

Diseñar **programas legibles** definiendo procedimientos, funciones, variables y parámetros para expresar las **subtareas y los datos** encontrados en el problema, asignándoles una **denominación representativa**.

Incorporar, al momento de pensar una solución, la posibilidad de **modificar un programa existente** para adecuarlo a problemas similares, y comprender que la legibilidad facilita las posibilidades de modificación.

Identificar **regularidades** en un programa y expresarlas a través de las herramientas adecuadas del lenguaje (por ejemplo, repeticiones, procedimientos y funciones y **parámetros**⁶³) para mejorar su legibilidad.

Desarrollar **criterios para evaluar soluciones** desde distintos puntos de vista (legibilidad, reutilización y eficiencia) teniendo en cuenta las necesidades del problema.

Extender la noción de programa **correcto** a sus procedimientos, funciones o **módulos**.

Contenidos → **B1.2. Algoritmos y estructura de datos: problemas clásicos, complejidad y recursión.** 

Alcances ↓

Secundaria, ciclo básico

--

Secundaria, ciclo orientado

Comprender que un **algoritmo** es una **solución general** que funciona para todas las instancias de un problema en forma automática, reconociendo la existencia de diferentes formas algorítmicas (dividir y conquistar, algoritmos golosos, programación dinámica, algoritmos heurísticos, etc.).

Reconocer la necesidad de la **generalidad** al momento de pensar una solución de forma algorítmica (cómo elaborar una solución que funcione para todas las instancias de un mismo problema).

Reconocer que el **diseño de un algoritmo está limitado** a las posibilidades de ejecución en una computadora (tanto respecto de sus formas de operación como respecto de la cantidad de datos a procesar).

⁶³ Ver el eje B3. Lenguajes de programación.

Comprender las **estructuras de datos** como formas de organizar los datos desde el punto de vista de las operaciones que se requieren para acceder a ellos, considerando la eficiencia de los accesos en forma general (por ejemplo, no es lo mismo tener datos desordenados que ordenados, no es lo mismo acceso lineal que binario, etc.).

Conocer la noción de **complejidad temporal y espacial**⁶⁵ de un algoritmo, y utilizarla para comparar la eficiencia de diferentes algoritmos.

Comprender la influencia de la manera de **organización de los datos** (o sea, de las estructuras de datos) en la **definición y funcionamiento de algoritmos** efectivos y en el costo final de los mismos (por ejemplo, árboles binarios para realizar búsquedas binarias).

Conocer la noción de **recursión** como una forma de resolver problemas computacionales basándose en instancias más pequeñas del mismo problema (por ejemplo, ordenar una secuencia de elementos dividiéndola en dos partes iguales, ordenarlas de la misma manera y luego combinar las dos soluciones en el resultado final).

⁶⁵ La complejidad de un algoritmo es una medida estimativa del consumo esperado de recursos del algoritmo al ejecutarlo en una entrada desconocida. Se habla de complejidad temporal cuando se hace referencia a la cantidad de trabajo requerida (usualmente vinculada al tiempo que demora el cálculo) y de complejidad espacial cuando se hace referencia a la cantidad de información que se debe manipular (usualmente vinculada al espacio de almacenamiento necesario).

Existen numerosas formas de calcular la complejidad. La más sencilla es contar la cantidad de milisegundos o bytes requeridos por una instancia particular, aunque es una medida poco fiel para casos genéricos, por lo que existen formas más sofisticadas que generalizan muchos casos particulares.

Contenidos → B1.3. Teoría de la computación: factibilidad y computabilidad. 

Alcances ↓

Secundaria, ciclo básico

--

Secundaria, ciclo orientado

Entender la noción de **computabilidad** de un problema como la existencia de una solución algorítmica para resolverlo.

Identificar que existe un **límite** para los problemas que se pueden resolver con una computadora y conocer **problemas no computables**.

Entender la noción de **factibilidad** de un algoritmo como un **límite** sobre su complejidad.

Conocer que existen **problemas inherentemente difíciles** (es decir, problemas computables, pero para los cuales no se conoce una solución factible).

Conocer la existencia de una clasificación de problemas según su **complejidad** (clases P, NP, NP-Completo y las relaciones entre ellas).

Eje → B2. Representación de información en la resolución de problemas computacionales 

Saberes ↓

Secundaria, ciclo básico

El reconocimiento de la existencia de **capas de representación** que establecen diferentes **niveles de abstracción** para utilizar diferentes herramientas del lenguaje en la definición de nuevas representaciones (mediante tipos básicos⁶⁶, registros⁶⁷ y listas) y modelizar la información de diferentes dominios cotidianos⁶⁸.

La comprensión de la computación como el **estudio del procesamiento de la información**, entendiendo que toda la información debe ser representada para poder procesarla.

Secundaria, ciclo orientado

La comprensión del papel que juegan **la eficiencia y la legibilidad** en el diseño de las representaciones que se utilizan incluyendo la compresión y la criptografía.

La **modelización de información** de diferentes dominios cotidianos mediante herramientas de los lenguajes de programación (como tipos básicos, registros, listas y sus combinaciones).

Contenidos → B2.1. Nociones de representación de información. 

Alcances ↓

Secundaria, ciclo básico

Comprender que la **información de un sistema computacional** requiere ser representada mediante elementos del sistema (por ejemplo, números, cadenas de

Secundaria, ciclo orientado

Reconocer las **herramientas de lenguajes de programación** que permiten expresar información (sensores, expresiones literales y funciones sobre tipos de datos tales como listas y registros para modelar entidades y operaciones del mundo cotidiano) y utilizarlas para construir transformaciones de información (sensores que combinan

⁶⁶ Los tipos básicos de un lenguaje de programación son aquellos que ya vienen contruidos con el lenguaje. Usualmente incluyen a los números y los booleanos, y pueden incluir otros como cadenas de caracteres (*strings*), colores, direcciones, etc.

⁶⁷ Ver el eje B3 de Lenguajes de programación.

⁶⁸ Un dominio cotidiano es el que se encuentra todos los días, como puede ser una red social, la agenda del celular, la lista de asistencia del aula, etc.

caracteres, cadenas de bits, etc.) para poder procesarse, y que el procesamiento consiste en la manipulación de dichas representaciones.

Comprender que la **computación** es el estudio del procesamiento de la información y que al utilizar cualquier artefacto computacional estarán manipulando datos.

Reconocer la existencia de representaciones de entidades mediante otras representaciones, que constituyen una serie de **capas de representación** (por ejemplo, un video se representa como una secuencia de imágenes, y esas imágenes se representan como secuencias de colores, y los colores como grupos de números y finalmente los números se pueden representar de forma binaria a través del uso de corriente en una computadora).

Reconocer las **herramientas de lenguajes** de programación que permiten expresar información (sensores, expresiones literales y funciones) y utilizarlas para construir transformaciones de información (sensores que combinan otros sensores y condiciones básicas sobre representaciones elementales, como comparación de valores).

Comprender que las transformaciones posibles de la información están vinculadas al **tipo de información** y relacionarlo con la noción de tipo de datos en los lenguajes de programación.

Reconocer diferencias entre la **representación digital y la analógica**⁶⁹.

otros sensores y condiciones básicas sobre representaciones elementales, como comparación de valores).

Comprender que la misma información se puede representar de diferentes maneras y ello impacta en la **eficiencia** (cantidad de elementos y de tiempo de cómputo, viabilidad de las transformaciones) y en la **legibilidad** (caracterización de la entidad a representar⁷⁰).

Vincular las formas de **representación de bajo nivel** (numeración binaria) con la idea de representación digital y con las características de eficiencia, legibilidad y las capas de representación vistas, y comprender las características que la hacen una buena representación (eficiencia en la cantidad de símbolos utilizados haciendo factible su implementación física en gran escala).

Reconocer la **compresión** como una manera de representar que requiere menos símbolos para representar la misma información.

Reconocer la **criptografía** como maneras de representar cuya interpretación es difícil sin tener las claves necesarias incluso por sistemas computacionales.

⁶⁹ Las representaciones analógicas utilizan una forma continua de señal como forma de codificación; estas representaciones usualmente están relacionadas con alguna forma física de representar la información. Las representaciones digitales, en cambio, emplean valores discretos (por ejemplo, una cantidad fija de números) para codificar la información. Por ejemplo, una representación analógica del sonido representa a este como una variación continua y constante de fluctuaciones de voltaje de acuerdo a las variaciones de presión de aire hechas por el sonido. En cambio, una representación digital del mismo sonido emplea una cantidad discreta de valores fijos de voltaje medidos cada cierto intervalo muy corto de tiempo.

⁷⁰ Por ejemplo, una representación de números en forma unaria requiere mil símbolos para representar la cantidad mil, mientras que una representación posicional requiere solamente cuatro símbolos.

Contenidos → B2.2. **Representaciones estandarizadas (imágenes, sonido, texto y números).** 

Alcances ↓

Secundaria, ciclo básico

Comprender la noción de **estándar de representación** y conocer los más frecuentes para las entidades abordadas (imágenes como mapa de bits tales como BMP o GIF, caracteres a través de formas de codificación de caracteres tales como ASCII o UNICODE, sonidos como secuencias de muestras —PCM— tales como WAV o AIFF).

Secundaria, ciclo orientado

Conocer **formatos comprimidos** para las entidades abordadas (por ejemplo, H265, JPG/PNG, MP3/FLACC) y comprender que están diseñados para un uso más eficiente de los recursos.

Valorar los **estándares abiertos** por sobre los cerrados en términos de distribución y acceso.

Contenidos → B2.3. **Modelado: representación con datos estructurados.** 

Alcances ↓

Secundaria, ciclo básico

Asociar la práctica de modelado con la **identificación de atributos** que sean significativos para el problema, de las **entidades** que se quieren expresar en el programa⁷¹.

Reconocer los **datos con estructura**, en particular, registros definidos por otras personas como modelos que representan entidades del mundo cotidiano.

Inferir las entidades representadas en modelos de datos conformados por registros.

Secundaria, ciclo orientado

Definir **registros** para modelar entidades con atributos.

Combinar **registros** y **listas** para modelar secuencias y conjuntos de entidades con atributos.

⁷¹ Por ejemplo, si la entidad considerada es una persona, sus atributos podrían ser el género, la edad, la profesión, el lugar de residencia, el nombre, el número de DNI, etc.

Interactuar con **registros** de datos para operar con las entidades que representan.⁷²

Identificar situaciones en las que hay entidades que se pueden **representar con listas**.⁷³

Comprender que cuando un programa trabaja con secuencias trata los elementos de a uno por vez, a diferencia de las personas, que obtenemos información de una secuencia considerando todos los elementos al mismo tiempo.⁷⁴

Utilizar **listas** para modelar **secuencias o conjuntos** de entidades.

Eje → B3. Lenguajes de programación

Saberes ↓

Secundaria, ciclo básico

La comprensión de **conceptos generales** de lenguajes de programación y la construcción de programas sumando las herramientas propuestas para este ciclo (parámetros, funciones, uso de registros, objetos y definición de listas) y la transferencia de conceptos entre **diferentes lenguajes**.

Secundaria, ciclo orientado

La comprensión de **conceptos generales** de lenguajes de programación y la construcción de programas sumando las herramientas propuestas para este ciclo (definición de registros, clases, bibliotecas y servicios⁷⁵), la transferencia de conceptos entre **diferentes lenguajes** y una aproximación a **herramientas de alto nivel** (otros paradigmas, funciones de alto orden).

⁷² Por ejemplo, para modelar una persona como registro sus atributos podrían ser el nombre, número de documento de identidad, edad.

⁷³ Por ejemplo, la lista de estudiantes modelados como una lista de registros con nombre y cantidad de días que faltaron o un diccionario del lenguaje que se representa con una lista de asociaciones entre palabras y definiciones.

⁷⁴ Por ejemplo, al mirar una mano de naipes podemos determinar si son todas del mismo palo o no.

⁷⁵ Un servicio es una aplicación que realiza tareas automatizadas y brinda cierta información en forma remota a otros programas (por ejemplo, conocer la hora oficial o el valor presente del tipo de cambio entre dos monedas).

Contenidos → B3.1. La sintaxis como un aspecto de los lenguajes de programación en bloques y texto. 

Alcances ↓

Secundaria, ciclo básico

Reconocer que existen **reglas** para la combinación de comandos y expresiones que determinan cuáles son las **combinaciones válidas** tanto en lenguajes bloques como en lenguajes textuales.

Secundaria, ciclo orientado

Conceptualizar la **sintaxis** como las reglas de combinación de comandos y expresiones que determinan las combinaciones válidas.

Comprender que la sintaxis en lenguajes de bloques está dada por la forma de los bloques, por lo que es muy difícil encontrar errores de sintaxis.

Comprender que en los lenguajes textuales aparece una nueva categoría de **errores asociados a la sintaxis**.

Utilizar adecuadamente los símbolos según la sintaxis del lenguaje para construir programas utilizando texto.

Contenidos → B3.2. **La semántica como el significado de los programas y sus partes en términos del problema que resuelve.** 

Alcances ↓

Secundaria, ciclo básico

Anticipar qué **transformación de información** expresa el uso de una **función** con argumentos⁷⁶ en un entorno de enseñanza de programación sin necesidad de ejecutarla.

Construir **el significado de una función con parámetros** como una transformación de información (la información de los argumentos —datos de entrada— se transforma en el resultado de la función —datos de salida—), distinguiéndolo de las operaciones individuales involucradas en la implementación de dicha transformación⁷⁷.

Comprender que la definición de una **función** incorpora un **nuevo operador al lenguaje**, y que el uso de ese operador aplicado a argumentos específicos es diferente de la definición de la función.

Interpretar la **aplicación de una función** a argumentos específicos como un **valor** (resultado de la operación) y no como el proceso involucrado en su cálculo. Por ejemplo, interpretar “el promedio de mis notas” como un número y no como un proceso de cálculo.

Identificar los **procedimientos** como representación de subtareas en el mundo de los **comandos** y las **funciones** como representación de subproblemas en el mundo de las **expresiones**.

Secundaria, ciclo orientado

Conceptualizar la semántica de un programa como el significado en términos del problema que resuelve.

Identificar que dos programas que resuelven el mismo problema son **equivalentes semánticamente**, aún cuando su sintaxis sea diferente o incluso estén escritos en diferentes lenguajes.

Apreciar la importancia de comprender los programas desde un punto de vista semántico para saber lo que resuelven y poder reutilizarlos en diferentes contextos y corregir sus errores.

Reconocer el rol de la **documentación** en la tarea de programar para facilitar y completar el aspecto comunicacional de los programas.

Reconocer la noción de **nivel de lenguaje**, como un criterio de comparación entre lenguajes de programación.⁷⁸

Reconocer una **clase** como un molde que permite construir objetos distintos con el mismo comportamiento.

⁷⁶ El uso de función con argumentos es un caso de utilización de una función (ver el eje B3. Lenguajes de programación) que toma datos específicos para sus parámetros, y que describe el resultado de transformar esos datos en el resultado, según la definición dada.

⁷⁷ Por ejemplo, el contraste entre transformar la información de notas obtenidas en diferentes materias en el promedio de dichas notas y sumar las notas de una y dividir por la cantidad de notas sumadas.

⁷⁸ Decimos que un lenguaje es más **imperativo o de más bajo nivel** que otro porque se parece más al lenguaje de máquina y que es **declarativo o de más alto nivel** porque permite expresar la solución al problema utilizando abstracciones cercanas al dominio del problema, en forma independiente de la arquitectura de la computadora.

Conceptualizar la noción de **estado** como un conjunto de valores que caracteriza un instante de la ejecución de un programa (por ejemplo en un videojuego, el nivel, las vidas, el tiempo restante, la posición de los personajes y los enemigos caracterizan un momento particular de la ejecución y eso conforman un estado).

Comprender que la solución de un problema computacional expresada por un programa se construye a partir de la **combinación de transformaciones de estado y transformaciones de información**.

Interpretar los **objetos** como la representación de una entidad que tiene un **estado** y un **comportamiento** asociados (por ejemplo, todos los personajes se mueven para cuatro lados, tienen posición y color).

Contenidos → B3.3. Herramientas de lenguaje de programación.

Alcances ↓

Secundaria, ciclo básico

Crear programas en entornos de enseñanza de programación por bloques o texto que combinen comandos, expresiones y eventos para resolver problemas.

- Utilizar **comandos** primitivos, secuencias, alternativas condicionales, repeticiones simples, repeticiones condicionales, repeticiones indexadas⁷⁹ y procedimientos en la construcción de programas.

Secundaria, ciclo orientado

Crear programas en entornos de enseñanza de programación en texto o en lenguajes textuales que combinen comandos, expresiones, eventos y definición de clases para resolver problemas.

- Utilizar **comandos** primitivos, secuencias, alternativas condicionales, repeticiones simples, repeticiones condicionales, repeticiones indexadas y procedimientos en la construcción de programas.

⁷⁹ Una repetición indexada es una forma de repetición que utiliza un índice para repetir la secuencia de comandos indicada para cada elemento de una lista dada (en forma explícita o implícita).

- Utilizar **expresiones** literales (números, cadenas de caracteres y booleanos⁸⁰), sensores, parámetros, variables⁸¹, registros, listas, funciones y operadores simples en la construcción de programas.
- Utilizar **eventos** de interacción con el usuario o el entorno (por ejemplo, para responder al uso de teclas o del *mouse* o responder a un *hardware* específico como el reloj o el acelerómetro) para definir programas interactivos.

Comprender que las herramientas de lenguajes de programación aprendidas son **independientes del lenguaje** específico utilizado o del “estado del arte” de la tecnología.

Comprender que muchas operaciones primitivas (tanto comandos como expresiones) requieren que se les suministre información para su funcionamiento, y que la forma de hacerlo es a través del uso de expresiones en forma de **argumentos** en su invocación.

Definir y utilizar **funciones**⁸² para expresar de manera sencilla y legible expresiones más complejas.

Identificar la necesidad y utilidad de los **parámetros**⁸³ como formas de **proveer información** en la forma de argumentos a los procedimientos y funciones que se definen.

Incorporar **parámetros** a la definición de procedimientos y funciones de manera de expresar **soluciones generales** a problemas con mayor número de instancias mediante un único procedimiento o función.

- Utilizar **expresiones** literales (números, cadenas y booleanos), sensores, parámetros, variables, registros, listas, funciones y operadores simples en la construcción de programas.
- Utilizar **eventos** de interacción con el usuario o entorno (por ejemplo, para responder al uso de teclas o del *mouse* o responder a un *hardware* específico como el reloj o el acelerómetro) para definir programas interactivos.

Reconocer que las **bibliotecas** son una manera de agregar funcionalidades a las herramientas básicas de un lenguaje, incorporarlas a sus programas e invocar las operaciones que proveen.

Utilizar algún lenguaje de **programación textual**⁸⁶, para resolver problemas computacionales en entornos que no son de enseñanza, transponiendo las herramientas ya aprendidas.

Comprender que los **conceptos de programación** aprendidos no están ligados a un lenguaje en particular y que, por lo tanto, son **independientes del lenguaje** específico utilizado o del “estado del arte” de la tecnología y ser capaces de reconocerlas en diferentes lenguajes, para poder **explorar nuevos lenguajes** por cuenta propia.

Identificar los distintos **alcances de variables**⁸⁷ (locales o globales) en diferentes lenguajes de programación y su aplicación en los lenguajes utilizados.

Adquirir fluidez en la utilización de **tipos estructurados de datos** (registros, listas y sus combinaciones).

Reconocer la importancia de construir programas que sean entendibles por las personas, y utilizar las herramientas brindadas por el lenguaje de programación para conseguirlo (en particular, uso de procedimientos y funciones para la división en

⁸⁰ El tipo booleano es un tipo de datos que representa valores de verdad. Los valores booleanos son dos: verdadero y falso. El nombre de este tipo de datos se basa en el nombre del matemático que estudió la matemática asociada a los valores de verdad y sus operaciones: George Boole. Este tipo de datos se utiliza en alternativas y repeticiones condicionales como forma de tomar decisiones (cuál alternativa elegir, o si continuar repitiendo o frenar).

⁸¹ Una variable de un lenguaje de programación es un mecanismo para recordar información a través de un nombre. En algunos lenguajes estos nombres se asocian con posiciones de memoria, mientras que en otros tienen más similitudes con las variables de un problema matemático.

⁸² Una función es una herramienta de lenguajes de programación que permite definir una nueva expresión en el lenguaje.

⁸³ Un parámetro en un procedimiento o función es un agujero al que se le da nombre, y que espera se provea un cierto dato (el argumento) que se debe proveer al usar la operación, para poder completar la definición y funcionar como se espera. Por ejemplo, en la operación de EscribirNombre, los parámetros podrían ser nombreAEscribir y colorAUtilizar.

⁸⁶ En este ciclo se propone utilizar un lenguaje textual como una forma de acercar a las y los estudiantes a experiencias de programación fuera de entornos didácticos.

⁸⁷ El alcance de una variable es la parte del programa donde el nombre de la variable tiene sentido y puede ser utilizado para recordar el valor asignado a la misma.

Reflexionar sobre la utilización de procedimientos y funciones con parámetros como forma de proveer mecanismos de **reutilización** y **modularización**.

Definir y utilizar **procedimientos y funciones con y sin parámetros** para plasmar estrategias de solución en el programa explicitando los subproblemas.

Reconocer que las **expresiones** de un programa tienen un **tipo de dato**⁸⁴ (por ejemplo, números, letras, colores, cadenas, booleanos, registros) que está relacionado con cómo se puede operar con ellas (por ejemplo, las expresiones de tipo “número” se pueden sumar, las expresiones de tipo “cadena” se pueden concatenar, las expresiones de tipo “booleano” se pueden combinar mediante negaciones, conjunciones y disyunciones, y los registros se pueden acceder con observadores de campo —por ejemplo, *damePalo* para los naipes—).

Considerar los **tipos de datos** involucrados en los argumentos de un comando o expresión y en los parámetros de procedimientos y funciones para interpretar y solucionar **errores de ejecución**.

Anticipar qué tipo de dato es necesario en una expresión a partir de la naturaleza de un operador.

Distinguir entre **datos simples** y **datos compuestos** que modelen entidades individuales (como un naipe o una persona), e identificar los componentes de un dato compuesto (por ejemplo, naipe —con número y palo—, personas —con nombre, apellido y otros datos—, etc.).

Utilizar **tipos estructurados de datos** como los **registros**⁸⁵ que modelen entidades mediante datos compuestos.

Reconocer la importancia de construir programas que sean entendibles por las personas, y utilizar las herramientas brindadas por el lenguaje de programación para conseguirlo (en particular, uso de procedimientos y funciones para la división en

subtareas, definición de tipos de datos adecuados —incluyendo secuencias—, y denominación representativa de los identificadores).

Conocer las herramientas propias de diferentes **paradigmas de lenguajes** y cómo influyen en el modo de expresar las soluciones en cada uno, para ampliar la noción de programa⁸⁸ más allá de la secuencia de comandos.

Conocer las herramientas de **alto orden** que brinda un lenguaje y utilizarlas en sus programas.

Definir una **clase** para capturar atributos y comportamientos comunes de los futuros objetos.

Reconocer que una forma de crear un **objeto** es instanciarlo⁸⁹ a partir de una **clase**.

⁸⁴ El tipo de una expresión es la denominación del conjunto al que pertenece la información descrita por dicha expresión.

⁸⁵ Los registros son una clase de tipos de datos de los lenguajes de programación cuyos elementos son datos estructurados que consisten en una colección fija de campos que describen varios otros datos más simples de diferentes tipos. Se puede acceder a los diferentes campos de un registro usando el nombre del campo sobre dicho registro.

⁸⁸ Por ejemplo, en un lenguaje de objetos un programa es un conjunto de objetos que resuelven un problema enviando mensajes. En un lenguaje lógico un programa es una base de conocimiento y un conjunto de reglas que resuelven un problema a través de una consulta a esa base.

⁸⁹ Una instancia es una copia concreta de ese molde con valores específicos para sus atributos.

subtareas, elección de tipos de datos adecuados y denominación representativa de los identificadores).

Ampliar la noción de dato compuesto para incluir datos que son **secuencias y conjuntos de datos**.

Utilizar **tipos estructurados de datos**, que representen **conjuntos y secuencias** de datos (por ejemplo, listas), y comandos y expresiones para recorrerlos (por ejemplo, `foreach`), modificarlos y acceder a sus elementos.

Utilizar herramientas de **repetición indexada** (por ejemplo, `foreach`, iteradores, etc.) para expresar las estrategias de recorrido de secuencias y listas.

Utilizar expresiones y comandos para agregar, acceder y eliminar **datos persistentes**.

Utilizar y modificar el **comportamiento** y **estado** de los **objetos**.

Reconocer que al utilizar **objetos** el comportamiento está en el objeto.

Contenidos → B3.4. Ejecución secuencial y paralela de programas.

Alcances ↓

Secundaria, ciclo básico

Reconocer que existen **modelos de ejecución no secuenciales** como el **concurrente** o **paralelo** y aplicaciones que utilizan estos modelos (tales como los personajes de una animación que ejecutan código en forma concurrente, juegos MMORPGs —Massively Multiplayer Online Role Playing Games— donde los jugadores llevan diferentes procesos en paralelo, cajeros automáticos corriendo en paralelo sobre la misma base de información del banco, documentos colaborativos en línea).

Secundaria, ciclo orientado

Comprender que las aplicaciones actuales (en la web o en los celulares) se basan en modelos de ejecución paralela en donde interactúan servicios y componentes de diversos orígenes.

Reconocer los **eventos** como la herramienta que permite la ejecución de un fragmento del programa asociada a alguna ocurrencia particular en el entorno (por ejemplo, se presiona una tecla, se recibe un mensaje, se cambia la orientación del celular).

Contenidos → B3.5. **Compilación e interpretación: intermediación entre el código fuente y la ejecución.** 

Alcances ↓

Secundaria, ciclo básico

Construir la idea de **código fuente** como la entidad que se construye (en términos de bloques o texto) y se puede guardar en un archivo y compartir.

Reconocer que para poder ejecutar un código fuente existe un proceso de intermediación que puede ser dado por un **intérprete** o un **compilador** propio de cada lenguaje de programación.

Construir la idea de **código ejecutable** como aquel que puede ser ejecutado efectivamente por una máquina específica sin ningún proceso de intermediación (por ejemplo, el archivo .exe).

Entender la **compilación** como la transformación del código fuente de un programa construido en un lenguaje en el código fuente de un programa construido en otro lenguaje, usualmente en términos de código ejecutable.

Secundaria, ciclo orientado

Comprender la diferencia entre el **proceso de compilación** (que dado un código fuente de un lenguaje produce un código fuente en otro lenguaje) y la **ejecución de código en forma interpretada** (donde hay un programa intermediario que va analizando y ejecutando cada instrucción de a una por vez).

Comprender la necesidad de que exista al menos un programa escrito en **código ejecutable** de la máquina para cada tipo de máquina que se quiera utilizar (puede ser únicamente un intérprete, por ejemplo, la Máquina Virtual de Java), y que ese programa se puede escribir directamente en el lenguaje de la máquina, o a través de un proceso de compilación.

Área → **C. Infraestructura tecnológica** 

Eje → **C1. Organización y arquitectura de computadoras** 

Saberes ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

La comprensión del **modelo de Von Neumann** como una forma general de organizar una máquina programable, común a la gran mayoría de computadoras, y el reconocimiento del rol de los componentes fundamentales para reconstruir la ejecución de un programa hasta la ejecución de las instrucciones en el procesador.

El conocimiento de las **especificaciones técnicas** de los componentes de *hardware* para comprender su relación con el desempeño de la computadora y poder elegir componentes en función de sus necesidades de desempeño.

--

Contenidos → **C1.1. La computadora como un sistema integrado por hardware y software.** 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Reconocer que las **soluciones computacionales** pueden implementarse tanto mediante **software** como mediante **hardware**, o combinaciones de ambas, así como reconocer beneficios y limitaciones de cada una.

--

Reconocer que el *software* tiene **necesidades específicas de hardware** y saber anticiparlas.

Identificar cómo participan, se relacionan y comunican los diferentes **componentes** al realizar **tareas específicas** (qué pasa en el sistema al encenderse, al ejecutar un programa, cuando algo falla, etc.).

Contenidos → C1.2. **Modelo de máquina programable: arquitectura von Neumann.** 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Completar la noción de computadora con el **modelo de arquitectura clásica de von Neumann**.

Entender cómo los componentes de la arquitectura clásica de von Neumann participan en la **ejecución de un programa** (la CPU busca datos e instrucciones de programas en la memoria, los procesa y los vuelve a guardar en la memoria).

Reconocer la **vigencia** de la arquitectura de von Neumann en los dispositivos computacionales que utilizan a diario.

--

Contenidos → C1.3. Componentes de *hardware*. 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Vincular los componentes fundamentales de *hardware* con el rol que ocupan en el **modelo de von Neumann** y cómo se comunican entre sí.

Conocer e interpretar las **especificaciones técnicas** básicas de los componentes fundamentales.

Reconocer que el mayor tiempo de procesamiento está en leer y escribir en la memoria y en los discos y la función de la **memoria caché** para amortiguar ese problema.

Comprender que contar con un procesador que tiene **varios núcleos** habilita posibilidades de procesamiento en paralelo, lo cual permite mejorar la velocidad del sistema cuando realiza muchas tareas en simultáneo.

Distinguir diferentes escenarios de decisiones a la hora de **seleccionar una computadora** que involucren necesidades de usuario, costos, funcionalidades, velocidad, almacenamiento.⁹⁰

Incorporar **vocabulario específico** para poder comunicarse con profesionales (servicio técnico o sitios de venta), participar en foros dedicados al análisis de dispositivos computacionales y de solución de problemas y leer especificaciones o manuales técnicos.

--

⁹⁰ Por ejemplo, una computadora para juegos necesita contar con mayor capacidad de cómputo y memoria, por lo que tendrá un precio alto; por otra parte, una computadora para juegos con mucho poder de cómputo, pero poca memoria, implica un balance incorrecto y un comportamiento no esperado.

Contenidos → C1.4. **Ejecución a bajo nivel: lenguaje de máquina, representación binaria e impulsos eléctricos.** 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Comprender el **lenguaje de máquina** como un lenguaje capaz de ser interpretado directamente por el procesador, formado por un conjunto reducido de instrucciones simples.

Comprender que los programas creados en cualquier lenguaje de programación tienen que ser **traducidos al lenguaje de máquina** para poder ser ejecutados por la computadora o deben poder ser interpretados por un programa que esté en ese lenguaje.

Comprender que el *hardware* se organiza basándose en **impulsos eléctricos**, y de ahí extrapolar la necesidad de representar la información con **dígitos binarios**.

Comprender que toda la información que se procesa y almacena en una computadora (por ejemplo, texto, imágenes o video) es representable a través de combinaciones de **unos y ceros** (que expresan los estados de presencia o ausencia de electricidad en un cable o magnetismo en diferentes componentes) y que, en definitiva, procesar dicha información consiste en un cómputo que realiza operaciones sobre estas representaciones.

Comprender cómo se relacionan las diferentes **partes del procesador** (contador de programa, ALU, unidad de control) y la memoria RAM al momento de ejecutar una instrucción del lenguaje de máquina (el **ciclo de instrucción**).

Comprender que los diferentes componentes están contruidos por **circuitos electrónicos** que permiten operar y almacenar la información representada con electricidad.

--

Eje → C2. Sistemas operativos

Saberes ↓

Secundaria, ciclo básico

El reconocimiento del sistema operativo (SO) como **administrador de recursos** de *hardware* y como **interfaz entre hardware y software** para dimensionar su injerencia en el funcionamiento del sistema.

El reconocimiento del rol del **sistema de archivos** para interactuar con los dispositivos de almacenamiento y el rol de los procesos para la ejecución de múltiples programas.

Secundaria, ciclo orientado

La construcción de una **visión de capas e interfaces** del sistema compuesta por el *hardware*, los controladores, el sistema operativo, las bibliotecas de sistema y el *software* de aplicación.

El reconocimiento de la relevancia de las **máquinas virtuales** como un entorno para proteger y compartir recursos de *hardware*, en los servicios de internet de uso contemporáneo.

Contenidos → C2.1. El sistema operativo como administrador de recursos e interfaz para el *hardware*.

Alcances ↓

Secundaria, ciclo básico

Reconocer al SO como una solución de **software que administra los recursos de hardware** entre los múltiples programas que se ejecutan en la computadora: planificación del uso del procesador, administración de memoria, de los dispositivos de entrada y salida, y de archivos, etc.

Comprender la importancia del sistema operativo como una **interfaz entre el hardware y el software** de manera que se pueda modificar un componente de *hardware* sin cambiar las aplicaciones de *software*, y viceversa.

Secundaria, ciclo orientado

Reconocer que existen **bibliotecas** que se usan en el desarrollo de programas de usuario para acceder a las **funcionalidades que el SO ofrece** (por ejemplo, interactuar con un componente de *hardware* en particular, reservar memoria, operar con archivos, etc.) y que estas bibliotecas constituyen una **interfaz** entre las aplicaciones y el SO.

Reconocer a los **controladores** como una **interfaz** entre el SO y el *hardware* que permite al SO interactuar con una variedad de dispositivos de *hardware* diferentes y que permite modificar el *hardware* sin tener que modificar el SO.

Comprender que un **controlador** es un **componente de software** que permite que el sistema operativo acceda a las funciones de un determinado *hardware*.

Reconocer que cada sistema operativo hace un **uso singular de los recursos de hardware**, que puede ser más o menos eficiente.

Contenidos → C2.2. **Abstracciones del sistema operativo: archivos y sistemas de archivos, procesos, virtualización, usuarios.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer la importancia del **sistema de archivos** en tanto permite interactuar con los **dispositivos de almacenamiento** viéndolos como **particiones** con **directorios** y **archivos**.

Reconocer que existen **diferentes sistemas de archivos**, y por defecto cada sistema operativo es **compatible** con algunos sistemas de archivos.

Reconocer que todo lo que se ejecuta en una computadora es parte de un **proceso** y que el comportamiento que vemos de un programa suele involucrar la ejecución de **múltiples procesos**.

Comprender que el SO se ocupa de **administrar los recursos** (espacio de memoria, tiempo de procesador, acceso a la red, etc.) que necesitan los procesos para ser ejecutados de forma simultánea.

Secundaria, ciclo orientado

Reconocer que una **máquina virtual** es una simulación de una computadora y, por lo tanto, los efectos sobre ella no impactan sobre el resto del sistema.

Reconocer las **máquinas virtuales** como un entorno de *software* que permite ejecutar sistemas operativos.

Reconocer la existencia de las máquinas virtuales como un **entorno** para proteger y compartir recursos de *hardware*, **aislando** por completo los programas que corren en las máquinas virtuales entre sí y de la máquina anfitriona (por ejemplo, en los servidores que soportan múltiples servicios o usuarios).

Eje → **C3. Redes e internet** 

Saberes ↓

Secundaria, ciclo básico

La comprensión de la **red internet como un modelo de capas** donde cada capa resuelve un problema y brinda una funcionalidad a la capa siguiente asociada a un protocolo.

El reconocimiento de diferentes tipos de modelos que permiten **distribuir archivos y tareas** a los diferentes dispositivos conectados (**modelos cliente-servidor y P2P**), sus ventajas y limitaciones.

La utilización de **herramientas de diagnóstico** para identificar problemas y conocer el estado de la red.

Secundaria, ciclo orientado

El reconocimiento de la existencia de aplicaciones que permiten controlar el **flujo** y la **anonimización** en el intercambio de información.

Contenidos → **C3.1. Estructura y funcionamiento de redes de computadoras. Internet como una red.** 

Alcances ↓

Secundaria, ciclo básico

Comprender que el diseño de internet es un **modelo de capas** donde cada una resuelve un problema específico y brinda una funcionalidad a la capa siguiente.

- Reconocer que en cualquier comunicación existe un **medio de transmisión** cableado (cable de cobre, fibra óptica) o un medio inalámbrico (ondas de radio) y que es utilizado directamente por la **capa física**.

Secundaria, ciclo orientado

--

- Reconocer los **enlaces de datos** como una manera de intercambiar información entre un conjunto acotado de computadoras que permite identificar pérdidas de información que pueden haber sucedido en la transmisión de la capa física.
- Reconocer que la **capa de red** agrega la posibilidad de intercambiar información entre un conjunto de computadoras de mayor escala (en distancia y en cantidad).
- Reconocer que la **capa de transporte** permite transmitir de manera confiable cantidades arbitrarias de información entre aplicaciones, resolviendo pérdidas y alteración del orden de la información propias de las limitaciones de la capa de red.
- Reconocer que la **capa de aplicación** contiene a los programas que utilizan la red para el intercambio de información.

Comprender la noción de **protocolo** como un conjunto de reglas de cómo deben comportarse los dispositivos y los artefactos individuales para garantizar un comportamiento particular del conjunto.

Asociar las funciones de cada **capa** del modelo con la existencia de un **protocolo** específico:

- Identificar que **Ethernet** y **wifi** son protocolos de la **capa de enlace** y, por lo tanto, son un conjunto de reglas que permiten utilizar medios físicos para intercambiar información entre un conjunto acotado de computadoras.
- Identificar que para que la **capa de red** funcione es necesario identificar las computadoras y retransmitir los paquetes de manera que lleguen a destino (**enrutamiento**) y que este problema se resuelve de manera distribuida y dinámica mediante el **protocolo IP**.
- Identificar que para que la **capa de transporte** cumpla su función se **fragmenta** la información en paquetes numerados del mismo tamaño y se define un **puerto** como un identificador que permite reconocer a qué programa debe entregarse la información; y que este problema se resuelve mediante protocolos de transporte, como **TCP** o **UDP**.
- Conocer que los programas de la **capa de aplicación** definen protocolos propios para su funcionamiento (por ejemplo, **HTTP** para páginas web y **BitTorrent** para el intercambio de archivos) que funcionan sobre los protocolos de transporte.

Valorar los **protocolos abiertos** porque permiten construir distintos *softwares* para las aplicaciones (por ejemplo, los protocolos abiertos de la web permiten que se construyan una variedad tanto de navegadores como servidores web, a diferencia de

las aplicaciones de mensajería privadas, donde solo podemos usar la aplicación que provee la empresa dueña de la plataforma).

Reconocer que dentro de una red privada los dispositivos se identifican con direcciones **IP privadas** y que, al acceder a internet, todos lo hacen con la misma **IP pública**.

Reconozcan que las **URL** incluyen una referencia a una computadora de la red mediante su identificación IP y que la traducción entre nombres de dominio y direcciones IP es realizada por los servicios de **DNS**.

Conocer las **métricas** más habituales para medir eficiencia en internet (**ancho de banda, latencia**, etc.) siendo capaces de interpretar en forma práctica algunos valores de tales medidas (por ejemplo, la diferencia entre una conexión de 100 Mbps y una velocidad de bajada de 20 Mbps).

Utilizar **herramientas de diagnóstico de redes** (por ejemplo, **ping, traceroute**) para conocer el estado de la red, identificar problemas y proponer soluciones.

Identificar que los grandes puntos de interconexión y los grandes enlaces pertenecen a un **grupo reducido de empresas** asociadas a países centrales y cómo esto impacta en la provisión del servicio y la **neutralidad de la red**.

Contenidos → C3.2. **Aplicaciones en internet: almacenamiento remoto, www, modelo cliente-servidor y P2P.** 

Alcances ↓

Secundaria, ciclo básico

Secundaria, ciclo orientado

Conceptualizar el **modelo cliente-servidor** como una forma de organizar las tareas de las aplicaciones utilizadas en internet.

Reconocer los roles de **cliente** y **servidor** en relación con las tareas que realiza cada uno para el funcionamiento de una aplicación (por ejemplo, en una aplicación de

--

mapas, el cliente brinda una interfaz para elegir un lugar, y el servidor se encarga de calcular y enviar la ruta).

Conocer el **modelo P2P** como una manera de organizar una aplicación en la que todas las computadoras pueden interactuar entre sí sin roles diferenciados.

Contrastar las **posibilidades de lucro y de control** (por ejemplo, del tráfico de datos, del servicio, de usuarios) entre el **modelo cliente-servidor** y el **modelo P2P**, en cuanto, en el primero, el control es unilateral y, en el segundo, el control se encuentra distribuido y es, por lo tanto, más complejo de direccionar.

Identificar que la existencia de las aplicaciones actuales de internet depende de **servidores** y grandes **centros de cómputos** que son propiedades de un **grupo reducido de empresas**.

Contenidos → **C3.3. Aplicaciones en internet: almacenamiento remoto, www, modelo cliente-servidor y P2P.** 

Alcances ↓

Secundaria, ciclo básico

Conocer que existen **protocolos seguros para la capa de transporte** y que, por lo tanto, pueden brindarle seguridad a las aplicaciones de la capa de aplicación sin tener que modificarlas (al agregar SSL/TLS sobre TCP, el protocolo HTTP se convierte en un protocolo seguro, que se suele llamar “HTTPS”).

Reconocer que los **protocolos seguros** de transporte (SSL/TLS) utilizan cadenas de **certificados** que permiten **autenticar** al sitio y **encriptar** la comunicación.

Tomar conciencia de que **al utilizar internet** siempre figura la dirección IP del usuario y que esto puede ser utilizado para identificarlo y geolocalizarlo, y que además, las aplicaciones pueden enviar información extra (por ejemplo, los navegadores envían

Secundaria, ciclo orientado

Conceptualizar las aplicaciones de **VPN** como una **simulación por software de una red privada** entre computadoras conectadas a una red pública.

Conocer que existen herramientas que intentan **anonimizar el uso de internet** (por ejemplo, TOR).

Conocer que existen herramientas que permiten **controlar el tráfico entrante y saliente** tanto de un dispositivo como de una red (por ejemplo, *Firewall* o cortafuegos).

información sobre el sistema operativo, modelo del navegador, tamaño de la pantalla, detalles de *hardware*, etc.) que también puede usarse para **identificar a un usuario**.

Conocer las **cookies** como una manera en que los sitios web almacenan información en nuestros dispositivos (por ejemplo, sesiones iniciadas o preferencias de búsqueda) y cómo pueden ser utilizadas para rastrear información de las acciones de un usuario en distintas páginas de internet.

Conocer la existencia de **redes privadas** como un mecanismo de seguridad que permite limitar quiénes participan de la red y vincularlo con el requerimiento de contraseña para conectarse a la red.

Área → **D. Datos e Inteligencia artificial** 

Eje → **D1. Recolección y modelado de datos** 

Saberes ↓

Secundaria, ciclo básico

El **modelado de entidades con atributos para recolectar datos** en manera activa que se almacenan en planillas, entendidas como forma elemental de base de datos.

Secundaria, ciclo orientado

La comprensión de las nociones básicas de **bases de datos** relacionales y no relacionales para almacenar y recuperar datos.

La valoración de los conjuntos de **datos abiertos** como una manera de compartir conocimiento.

Contenidos → **D1.1. Recolección de datos activa y pasiva.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer de qué manera los artefactos computacionales recolectan información, diferenciando cuándo **recolectan la información de modo activo o pasivo** (por ejemplo, solicitar información mediante un formulario o un programa que registre la información de sitios visitados).

Secundaria, ciclo orientado

Reconocer la existencia de conjuntos de **datos de acceso abierto** recolectados por diferentes organizaciones y valorarlos como una forma de **compartir conocimiento**.

Identificar si existen conjuntos de datos de acceso abierto que pueden ser utilizados para **analizar un determinado fenómeno**.

Diseñar **encuestas en línea** utilizando diferentes tipos de preguntas para recolectar datos en función de los **modelos** que definan para resolver una situación problemática.

Reflexionar sobre la importancia de **conocer cuándo se está realizando recolección pasiva** de datos y sobre el hecho de que no está permitido por ley **recolectar datos personales sin aviso**.

Reconocer que la recolección de datos necesita de un **almacenamiento físico** y dimensionar su **escala** en aplicaciones contemporáneas.

Reconocer que es posible obtener una variedad de **datos de la mayoría de fenómenos u objetos** y que para esto son necesarias herramientas de recolección.

Reconocer si es suficiente contar con una herramienta digital de recolección activa o se necesita contar con un programa que **automatice la recolección de datos** de forma pasiva.

Contenidos → D1.2. Modelado de datos: entidades con atributos. Base de datos.

Alcances ↓

Secundaria, ciclo básico

Reconocer que existen diferentes características (**atributos**) que nos permiten modelar diferentes elementos (**entidades**) para recolectar información relevante, almacenarla y analizarla.

Identificar entidades y atributos en situaciones contemporáneas en las que haya recolección de datos.

Proponer un **modelo de entidades con atributos** para recolectar datos.

Organizar los datos de **entidades con atributos en una planilla de cálculo** propia (por ejemplo, utilizando una fila para cada entidad y las columnas para cada atributo) e interpretar en estos términos una planilla con este tipo de datos generados por otras personas.

Secundaria, ciclo orientado

Utilizar un **lenguaje de consultas** para almacenar y recuperar información de una base de datos relacional o no relacional aplicando proyecciones y filtros a partir de una o más tablas.

Comprender las nociones de **tabla, fila y columna** en términos de **modelado de entidades con atributos**.

Comprender la noción de **identificador unívoco** como un atributo (o grupo de atributos) que permite identificar de manera única a una entidad en una base de datos relacional o no relacional (por ejemplo, el DNI o el CUIL en el caso de las personas, clave primaria en base de datos relacional) y su importancia para la recuperación de la información sobre esa entidad.

Reconocer en las planillas de cálculo una **forma elemental de base de datos**.

Aproximarse a la noción de **base de datos** a partir de reconocer en situaciones contemporáneas la presencia de conjuntos de datos almacenados y organizados cuyo procesamiento o registro son centrales para el funcionamiento de un artefacto computacional (por ejemplo, cuando se realiza una compra en un supermercado se consulta a la base para asociar un código con un precio).

Conocer los **índices** de una tabla como una estructura de datos auxiliar que permite la recuperación de información de manera mucho más eficiente.

Identificar los **motores de bases de datos** como la pieza de *software* que implementa bases de datos de forma eficiente, garantizando ciertas propiedades fundamentales de los datos (por ejemplo, consistencia, confidencialidad, acceso simultáneo de muchos usuarios) y que permite ejecutar las consultas y verificar permisos.

Implementar una **base de datos** en un motor de base de datos, para almacenar datos y recuperarlos.

Eje → D2. Ciencia de datos, inteligencia artificial y aprendizaje automático

Saberes ↓

Secundaria, ciclo básico

La aproximación a la noción de **inteligencia artificial (IA)** como un conjunto de técnicas para desarrollar **artefactos computacionales** que no son inteligentes, sino que **reproducen comportamientos humanos**.

El **desarrollo crítico de modelos de aprendizaje automático (AA)** que les permita profundizar los conceptos importantes en el proceso de desarrollo (conjunto de entrenamiento, entrenamiento, conjunto y métricas de evaluación, representatividad de los datos) y reflexionar sobre su impacto, teniendo en cuenta que estos artefactos reflejan subjetividades que pueden reproducir y profundizar desigualdades y prejuicios.

La aplicación de **herramientas de procesamiento automático de datos** (filtros, transformaciones y totalizaciones) para generar **visualizaciones** y **analizar** un conjunto de datos.

Secundaria, ciclo orientado

La aplicación de **herramientas de procesamiento y análisis automático de datos** tanto para la **preparación** de conjuntos de datos como para la evaluación de su **representatividad** antes de usarlos para extraer conclusiones o entrenar y evaluar modelos de aprendizaje automático.

El reconocimiento de las aplicaciones de inteligencia artificial como artefactos computacionales **fallibles**, para los que en general no es posible explicitar el criterio que define y guía su funcionamiento.

Contenidos → D2.1. **Aplicaciones de inteligencia artificial. Mecanismos generales.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer que la **IA** es una denominación para referirse a **artefactos computacionales** que buscan **imitar comportamientos humanos** a distintos niveles (por ejemplo, un predictor de texto vs. un modelo de lenguaje generativo) y que pueden estar contruidos con técnicas muy diversas (por ejemplo, sistemas de reglas vs. redes neuronales pre-entrenadas).

Reconocer que la IA tiene diferente grado de **aplicabilidad** según el tipo de problemas, y en qué tipo de aplicaciones y problemas es más frecuente encontrar soluciones exitosas basadas en modelos de aprendizaje automático (reconocimiento de imágenes, reconocimiento de voz o de imágenes, generación o traducción automática de texto, etc.).

Reconocer que el **comportamiento que se percibe como inteligente** en las aplicaciones de IA existe gracias a la **inteligencia de los humanos** (ya sea porque los humanos definen las relaciones lógicas o porque están basadas en ejemplos de humanos resolviendo el problema).

Reconocer que las aplicaciones de **AA** imitan el comportamiento humano que infieren a partir de un conjunto enorme de **ejemplos** y diferenciar esto de la forma en la que las personas resuelven un problema, utilizando su intuición, su experiencia, sus conocimientos, su razonamiento, su creatividad, etc.

Secundaria, ciclo orientado

Formar criterio para evaluar las respuestas que brindan las aplicaciones basadas en IA, contemplando desde una **mirada crítica** diversas variables (el conjunto de datos, el nivel de representación de una población, país, región).

Reconocer que las aplicaciones basadas en IA cometen **errores** y que su éxito se mide en métricas estadísticas personalizadas para cada solución, que no se encuentran consensuadas y que generalmente no contemplan el impacto ético.

Contenidos → D2.2. **Preparación, análisis y visualización de datos.** 

Alcances ↓

Secundaria, ciclo básico

Reconocer que para extraer conclusiones de un conjunto de datos es necesario **procesarlo** y **visualizarlo**.

Comprender que las **operaciones de filtrado** permiten seleccionar, del conjunto de datos, aquellos elementos que cumplen con una condición definida.

Comprender que las **operaciones de transformación** producen nuevos datos a partir de aplicar una transformación o una función a cada elemento de un conjunto de datos.

Comprender que las **operaciones de totalización** calculan un valor a partir de un conjunto de elementos (por ejemplo la sumatoria, el promedio, la cantidad, el máximo, reordenamientos, etc.).

Analizar conjuntos de datos utilizando combinaciones de estas herramientas para **responder preguntas y extraer conclusiones**.

Formar criterio para elegir el **tipo de gráfico** más adecuado a los propósitos de visualización y comunicación que se persiguen.

Reconocer que las **decisiones** que se toman durante el procesamiento de un conjunto, así como las **conclusiones** que se obtienen de su análisis, dependen de **interpretaciones subjetivas** de quienes lo analizan.

Reconocer que el diseño de las **herramientas de recolección** es importante para no generar errores en los datos recolectados (campos vacíos o errores de tipo de dato).

Secundaria, ciclo orientado

Reconocer la necesidad de la **preparación de los datos** como un primer paso de procesamiento (limpiar, recodificar, reordenar utilizando filtros, transformaciones y totalizaciones) para poder analizarlos.

Reparar problemas en un conjunto de datos (campos vacíos, erróneos, etc.).

Identificar durante el análisis de un conjunto de datos, **datos atípicos** o aislados (*outliers*) producto de errores de medición, así como su impacto en los resultados y la necesidad de filtrarlos.

Reconocer que las **conclusiones** a partir de la visualización y el análisis de datos no solo están afectadas por la **subjetividad** de quien interpreta los datos, sino también por la **representatividad** de la muestra del conjunto de datos.

Contenidos → D2.3. Modelos de aprendizaje automático. 

Alcances ↓

Secundaria, ciclo básico

Conceptualizar la **noción de clasificador** como un modelo de AA que toma un elemento del universo y le asigna una clase o una categoría (seleccionada de entre un conjunto definido previamente por una persona).

Identificar la existencia de clasificadores y sus clases en **aplicaciones de AA contemporáneas**.

Comprender que un **modelo entrenado** de AA es un programa que se obtiene como resultado de un **proceso automático de análisis de datos** (entrenamiento).

Reconocer que es posible generar modelos para resolver **problemas distintos** utilizando el **mismo entrenamiento y diferentes conjuntos de datos**.

Identificar que un **modelo sin entrenar** es un programa que tiene valores no definidos y que el objetivo del entrenamiento es analizar un conjunto de datos para definir estos valores y que esto se hace mediante un programa.

Reconocer que el **entrenamiento** de un modelo de AA infiere **patrones en los datos** que luego **reproducirá el modelo** entrenado al funcionar, pero que el funcionamiento obtenido no necesariamente refleja las intenciones de los desarrolladores.

Reconocer que para inferir el comportamiento efectivo de un modelo es necesario **ponerlo a prueba sobre datos nuevos**.

Reconocer que al diseñar un clasificador basado en AA para resolver un problema, tanto la acción de definir clases como la de clasificar elementos implica emitir un **juicio sobre el problema**.

Secundaria, ciclo orientado

Conceptualizar el **entrenamiento** de un modelo de AA como la **solución a un problema matemático** en el cual se buscan valores de parámetros para que una función de clasificación minimice su error sobre un conjunto de datos dados.

Reconocer que en general no es posible comprender a partir de analizar el código fuente de un modelo entrenado qué criterio utiliza para clasificar o predecir (**explicabilidad**) a diferencia de los programas de código abierto desarrollados enteramente por personas.

Comprender que la **significatividad y precisión** de las salidas de un modelo dependen de la cantidad, calidad y diversidad de los **datos recolectados**.

Conocer otros algoritmos de ciencia de datos que permiten clasificar y **predecir sin necesidad de entrenar** un modelo de IA (por ejemplo, clustering, KNN).

Preparar de forma automática un conjunto de datos para entrenar un modelo de AA, analizando posibles **sesgos** que puedan afectar la **representatividad** de la muestra.

Evaluar de **forma automática** un modelo de AA teniendo en cuenta distintos tipos de **métricas** y su interpretación (exactitud o *accuracy*, exhaustividad o *recall*, precisión o *precision*, F1, matriz de confusión) sobre un conjunto de datos de evaluación, teniendo en cuenta que no hay una de ellas que por sí sola garantice el buen funcionamiento.

Proponer ejemplos para poner a prueba modelos de AA que puedan **explicitar sus sesgos**.

Dimensionar la **cantidad de datos y de energía** que se requieren para entrenar estos modelos y vincularlo con el eje de Computación y ambiente y el eje de Redes e internet.

Generar críticamente un modelo de AA para resolver un problema:

- Anticipar impactos del modelo teniendo en cuenta el **contexto de uso**.
- Definir las **clases o categorías** a partir de analizar el problema.
- Conseguir **datos**, teniendo en cuenta las categorías definidas y la representatividad de la muestra.
- Definir el **conjunto de entrenamiento y de evaluación** teniendo en cuenta su **representatividad**.
- **Poner a prueba** el modelo sobre los datos de evaluación y tomar alguna métrica de desempeño (por ejemplo, exactitud o *accuracy*) teniendo en cuenta sus limitaciones.
- Evaluar el **impacto** del modelo funcionando en su contexto de uso.

Contenidos → D2.4. **Subjetividad expresada en los datos. Sesgo algorítmico.** 

Alcances ↓

Secundaria, ciclo básico

Comprender que las **aplicaciones de AA** son artefactos **creados por personas** basándose en conjuntos de datos recolectados por personas y, por lo tanto, están sujetos a las **subjetividades** de estas personas.

Comprender que todo **conjunto de datos** refleja un **recorte subjetivo** de la realidad, tanto en el diseño (qué se recolecta) como en la recolección (de qué manera y quiénes participan de la muestra).

Comprender que los conjuntos de datos al ser una muestra del mundo en un momento determinado pueden reflejar las **desigualdades existentes** y, en ese caso,

Secundaria, ciclo orientado

Comprender que un modelo de AA solo tendrá en cuenta aquello que figure en el conjunto de **datos de entrenamiento** y, por lo tanto, es fundamental que ese conjunto sea **representativo del fenómeno** que se quiere modelar y de la manera en la que se lo quiere modelar.

Comprender que las aplicaciones basadas en AA extraen generalizaciones de los conjuntos de datos a partir de encontrar regularidades y, por lo tanto, los subconjuntos de datos que reflejen **situaciones poco representadas** en el conjunto serán menos tenidos en cuenta.

si son usados para entrenar modelos, el comportamiento de esos modelos puede **repetir y reforzar** esas desigualdades.

Aproximarse a la noción de **representatividad estadística** de una muestra para identificar sesgos en conjuntos de datos.

Aproximarse a la noción de **sesgo algorítmico** como patrones de comportamiento de un artefacto computacional, y saber que puede estar introducido tanto en el diseño del programa como por los datos de entrenamiento.

Reconocer que las desigualdades que dan origen al sesgo algorítmico de un sistema de AA se refuerzan a partir de la **retroalimentación**.

Diferenciar **datos atípicos** o *outliers* de datos correctos sobre **poblaciones poco representadas**.

Encontrar relaciones entre la **subjetividad** que expresa un conjunto de datos con **cómo, para qué y por quiénes** los conjuntos de datos son organizados, recolectados, transformados y etiquetados.

Reconocer que la **subrepresentación** en los conjuntos de datos suele estar asociada a **relaciones de poder**.

Glosario de términos

→ Abstracciones

Se refiere a las construcciones y los modelos abstractos que expresan una solución genérica. Las abstracciones se desarrollan para crear soluciones computacionales. En programación, se instancian a través de diferentes [artefactos computacionales](#) (por ejemplo, procedimientos, funciones, bibliotecas de funciones, etc.). Denning (2004) define las abstracciones en el contexto de los comandos como: “objetos abstractos que realizan acciones”. En infraestructura, esta noción aplica a la noción de un dispositivo computacional construido a partir de componentes, pero cuya funcionalidad es posible verla como abstracta (lo que suele llamarse “caja negra”).

El uso de soluciones generalizadas y partes de soluciones diseñadas para su reutilización simplifica el proceso de desarrollo al brindar herramientas de más alto nivel para poder abordar la complejidad de un nuevo problema (K-12 Computer Science Framework Steering Committee, 2016: 78).

El modo en que se desarrollan las abstracciones es a través del modelado, la descomposición, la generalización y la clasificación.

El **modelado** consiste en la representación de un elemento, problema o situación del mundo real capturando los aspectos relevantes para un propósito particular y omitiendo cualquier otro detalle. Por ejemplo, una persona puede ser representada a través de su nombre, un número de documento de identidad y su fecha de nacimiento si el objetivo es construir un sistema de identificación unívoca a partir de datos registrados. Cualquier otro detalle de la persona (como el color de ojos, la estatura, etc.) no resultaría relevante para ese propósito. Otro ejemplo es la utilización de figuras geométricas para representar elementos de la realidad: modelar una casa a través del dibujo de la misma.

La **descomposición** consiste en dividir un problema complejo en problemas de menor complejidad, o sea, en subproblemas que sean fáciles de abordar y cuyas soluciones se puedan combinar para formar una solución general. A su vez, cada uno de los subproblemas puede ser descompuesto de la misma forma. Asimismo, esta capacidad también se puede entender en sentido contrario: para analizar un sistema, se puede analizar cada una de sus partes por separado, considerando las relaciones entre ellas y, finalmente, utilizar esa comprensión integradamente para el entendimiento del sistema en su totalidad. Ambos acercamientos son válidos para poder manejar la complejidad. Por ejemplo, la representación de una casa puede descomponerse en diversas partes: dibujar las paredes, el techo, las ventanas, la puerta, etc. A su vez, se puede descomponer el dibujo de la puerta: dibujar un rectángulo con un pequeño círculo para representar el picaporte.

La **generalización** es el reconocimiento de patrones comunes y la capacidad de dar nombres a los esquemas que expresan cada uno de esos patrones. Consiste en encontrar qué características son compartidas por una serie de elementos permitiendo explicitar cuáles los diferencian. Por ejemplo, si se desea construir un procedimiento que dibuje cuadrados de distintos tamaños, este deberá contemplar qué tienen en común los elementos. Así se podrá considerar si la forma de dibujarlos puede ser siempre la misma variando la extensión de los lados de las figuras. En este caso, “ser cuadrado” es la característica que los define y el tamaño es la que los diferencia. Esto permite, por ejemplo, en programación, definir una única tarea que dibuje cuadrados en general y que reciba como dato la extensión de los lados que el usuario quiere darle a cada cuadrado en particular. Así, el mismo código puede utilizarse en otras partes de un programa, pero estar escrito una única vez. Así también cualquier

modificación realizada sobre dicho código se verá reflejada en todos los sectores en donde es utilizado. Si no se generaliza la tarea, habría que definir una tarea distinta para dibujar cada cuadrado de distinto tamaño que se desee dibujar y frente a la necesidad de un cambio introducirlo en múltiples lugares del código.

La **clasificación** es entendida como la capacidad de agrupar un conjunto de elementos en diferentes categorías, o sea, definir clases o tipos. Es un caso especial de la generalización que se combina junto con el modelado porque se enfoca en las características comunes de una serie de elementos y, sobre esta base, construye una categoría o un tipo que representa exactamente estas características. Por ejemplo, si consideramos los elementos que constituyen el dibujo de una casa, estos pueden clasificarse según su forma geométrica: triángulos, rectángulos, trapecios, círculos, etc. A su vez, estas categorías pueden agruparse en el tipo de las figuras geométricas, que se distinguen de otras figuras, por ejemplo, de dibujo libre. Otra forma de razonar el proceso de clasificación es reconociendo qué operaciones pueden realizarse sobre una serie de elementos. Por ejemplo, los números pueden ser sumados, restados, multiplicados, divididos, entre otras operaciones, mientras que los colores pueden ser combinados de formas particulares para obtener nuevos colores. Por esta razón, se puede idear un tipo de elemento llamado “número” que represente la posibilidad de realización de las operaciones aritméticas y otro tipo de elemento llamado “color” que represente la posibilidad de realizar operaciones sobre colores.

→ Algoritmo

En el ámbito de las Ciencias de la Computación un algoritmo es una descripción general de un único proceso que permite construir una solución para todas las instancias de un problema, por ejemplo, la descripción de un método que permita transformar una lista cualquiera de números para que quede ordenada de menor a mayor. Esta concepción es mucho más sofisticada que una secuencia de pasos, por eso en esta propuesta, no consideramos como algoritmos los pasos para resolver tareas cotidianas (como atarse los cordones) y nos reservamos el término para introducir la noción disciplinariamente precisa en el último nivel de la escolaridad como parte del abordaje de algunos tópicos centrales en la teoría de la Computación.

Esta decisión es producto de discusiones del equipo en relación al término y los problemas que encadenó su uso simplificado o banalizado incluso en producciones previas de la propia Iniciativa Program.AR. Por un lado, no permite discutir la capacidad de un algoritmo de proponer una única solución general para resolver múltiples instancias de un problema y la complejidad que esto reviste al momento de su elaboración. Por ejemplo, en la formulación de un algoritmo de ordenamiento los “pasos” de la “secuencia” deben estar elegidos y ordenados de tal manera que resuelvan el problema sobre todas las listas posibles, mientras que en la serie de pasos para atarse los cordones no aparece la dificultad asociada a esta multiplicidad. Por otro lado, se rompe con el modelo de la computación como sistema o transformación de información: los pasos hablan de un proceso, pero ¿dónde está la información?, ¿dónde está la entrada?, ¿donde la transformación de información? ¿dónde la salida?. En definitiva, el problema de escribir secuencias de pasos para resolver tareas cotidianas esconde con su simplicidad estos aspectos que son inherentes a la noción disciplinaria, construyendo una definición no sólo equivocada de un término clave para la disciplina sino también insignificante para las y los estudiantes de nivel inicial y primaria pues introducida en esos niveles solo se utiliza para renombrar la noción de orden de pasos que ya conocen.

→ Artefacto computacional

Consiste en una construcción lógica –a veces en combinación con elementos físicos de distintos dispositivos– para solucionar problemas de naturaleza computacional. Algunos ejemplos de artefactos computacionales son los sistemas de *software* y programas específicos, los kits de robótica educativa, los protocolos de comunicación, las metodologías y herramientas vinculadas a soluciones computacionales, etc.

Se distinguen los artefactos de los [dispositivos computacionales](#), que se refieren al aparato físico destinado a llevar adelante cómputos: computadoras de escritorio, teléfonos inteligentes, televisores inteligentes, tabletas, consolas de videojuegos, sistemas de control de automóviles, artefactos hogareños o aparatos industriales, etc.

→ Ciencias de la Computación

Es el nombre del área del conocimiento que estudia las bases y los fundamentos sobre los que se articula la creación y utilización de tecnologías computacionales. Engloba un conjunto de saberes, tales como la programación, el funcionamiento de las redes de computadoras, la representación de la información, entre otros. Resulta fundamental para comprender un mundo cada vez más organizado y dependiente de la tecnología y para poder participar activa, libre y soberanamente en debates actuales sobre la interacción entre las computadoras y la sociedad.

→ Ciudadanía digital

Usualmente, el término denomina el comportamiento, las prácticas y las interacciones entre personas en entornos virtuales (por ejemplo, en salas de chats, juegos en línea, redes sociales, etc.). Frecuentemente, las propuestas de enseñanza referidas al término de “ciudadanía digital” abordan saberes relacionados a la seguridad y la ética informática brindando recomendaciones que dependen de elecciones y responsabilidades individuales. No brindan, por lo general, nociones claves respecto de cómo funcionan los sistemas y las estructuras macrosociales que permiten ataques y vulnerabilidades.

Nosotros proponemos un enfoque que, por un lado, tenga en cuenta que no hay escisión posible entre la ciudadanía digital y el ejercicio de nuestros derechos en otras esferas de la vida: lo que se ve afectado y transformado por la [tecnología digital y computacional](#) es el conjunto de derechos y obligaciones que tenemos como personas y también la cantidad y el tipo de delitos de los que podemos ser víctimas. Por otro lado, consideramos que, para poder tomar decisiones más o menos informadas y autónomas sobre el uso y modos de uso de dicha tecnología, es necesario conocer su funcionamiento.

→ Derechos de la ciudadanía asociados a los [artefactos y dispositivos computacionales](#)

Los derechos de la ciudadanía asociados a los artefactos y los dispositivos computacionales incluyen cuatro libertades básicas: la libertad de usar el artefacto o el dispositivo con cualquier propósito (uso), la libertad de estudiar cómo funciona el artefacto o el dispositivo y ser capaces de modificarlo adaptándolo a las necesidades propias (estudio), la libertad de distribuir copias del artefacto o dispositivo para ayudar a otros usuarios (distribución) y la libertad de mejorar el artefacto o el dispositivo y hacer públicas esas mejoras para beneficiar a la comunidad (mejora). Además, incluye la capacidad de hacer trabajo colaborativo para la construcción de nuevos artefactos y dispositivos, y se vincula con otras cuestiones tales como la propiedad intelectual, la vigilancia en las redes, la neutralidad en la red, circulación de información falsa, etc. Los primeros cuatro derechos se engloban bajo la denominación *software libre* (cuando se habla de artefactos) y *hardware libre* (cuando se habla de dispositivos), y se vinculan con las formas de licenciamiento de estos productos. Sin embargo, al enfocar el tema desde el área de ciudadanía digital, creemos que es mucho más rico hablar de

derechos asociados y hacer foco en ellos y en la capacidad de producción colaborativa. Se suele tomar al *software* y al *hardware* libres como un “tema” de estudio que comienza y termina. Nuestra propuesta, en cambio, es usar exclusivamente *software* libre en todas las instancias de la enseñanza para lograr la soberanía tanto pedagógica como tecnológica. Además, se forma al estudiantado con una visión colaborativa, solidaria, federal y nacional, un modelo antagónico al del individualismo, la mercantilidad, la ilegalidad y la dependencia que trae el uso de *software* privativo.

→ Dispositivo computacional

Denomina al aparato físico destinado a llevar adelante cómputos de diversa naturaleza (por ejemplo, computadoras de escritorio, teléfonos y televisores “inteligentes”, tabletas, consolas de videojuegos, robots, sistemas de control como los que pueden tener los automóviles, artefactos hogareños o aparatos industriales, etc.). Lo diferenciamos de los artefactos computacionales para distinguir la construcción lógica —a veces en combinación con elementos físicos— que se crea para solucionar problemas de naturaleza computacional, por ejemplo, sistemas de *software* y programas específicos, kits de robótica, protocolos de comunicación, metodologías y herramientas vinculadas a soluciones computacionales, etc. (ver [Artefacto computacional](#)).

→ Habilidades

Se utiliza como la aptitud o la capacidad de hacer una cosa, una actividad que se ejerce, un “saber hacer” general (Roegiers, 2000). Como señala Meirieu (1990): “ninguna capacidad existe en estado puro y toda capacidad se manifiesta a través de la aplicación de contenidos”. Por este motivo, no definimos el desarrollo de una habilidad por sí misma ni la aislamos como tal, ya que entendemos que se ejercitan o ponen en juego en la trama de contenidos específicos que le otorgan contexto (Maggio, 2018). Ejemplos de estas son: modelar, descomponer, generalizar, clasificar. (Se incluye una descripción de lo que implican estas habilidades en el contexto computacional a propósito de la definición de [abstracciones](#)).

→ Hardware libre

Ver [Derechos de la ciudadanía asociados a los artefactos y dispositivos computacionales](#).

→ Huella digital

Cuando uno navega en internet, se recolecta información sobre nuestra interacción (clics, sitios visitados, búsquedas, tiempo de permanencia, ubicación geográfica, etc.) en forma continua, eso constituye un rastro económicamente valioso que se conoce con el nombre de huella digital. La huella digital incluye datos que dejamos involuntariamente, además de los voluntarios.

→ Inteligencia artificial

En forma cotidiana y/o comercial se suele utilizar el término *Inteligencia Artificial* para denominar dos aspectos completamente diferentes: por un lado, el *aprendizaje automático* y la *ciencia de datos*, que es el aspecto más prolífico actualmente; y, por el otro, el estudio de la generación de *inteligencia* y *conciencia en forma artificial*, que actualmente está lejos de poderse concretar y constituye más una meta que una realidad. El término *inteligencia artificial*, en el imaginario popular, tiende a considerar a las máquinas actuales como entidades inteligentes o incluso pensantes, y esto no es (al día de hoy) correcto. Desde el ámbito educativo debemos elegir cuidadosamente la terminología para que cada fenómeno sea correctamente entendido y no utilizar el término más popular por “seguir la moda”. En esta propuesta, usamos los términos de *aprendizaje automático* y *ciencia de datos* cuando nos

referimos a la noción popularizada de inteligencia artificial. Esta decisión se basa en que no queremos contribuir a fomentar la equivocada idea de que las máquinas son entidades inteligentes.

→ Modelos de cómputo y modelos de arquitectura de computadoras

Un *modelo de cómputo* es una definición de un mecanismo abstracto que modela la ejecución automática de procesos de computación (Aho, 2011). Ejemplos de modelos de cómputo son las máquinas de Turing, las funciones recursivas, el cálculo lambda, los autómatas celulares, los sistemas de producción, las máquinas de acceso aleatorio (RAM, *random access machines*), entre muchos otros. Por otra parte, un *modelo de arquitectura* de computadoras es un conjunto de técnicas que permiten describir, implementar y construir máquinas programables (combinando diferentes componentes) que puedan ejecutar transformaciones de información en forma efectiva. El modelo de arquitectura más exitoso y difundido es el modelo von Neumann inspirado en el modelo de computación de las máquinas de Turing. La preeminencia del modelo de arquitectura von Neumann tuvo como consecuencia un foco centralizado en las formas de programación basadas en comandos (programación imperativa), relegando otras formas de expresar programas, como las expresiones. Al pensar la computación como una ciencia, debemos tener presente que el modelo de cómputo de las máquinas de Turing o de las máquinas de acceso aleatorio (que modelan la programación basada en comandos) es solamente una de las formas de expresar el procesamiento de la información (si bien una extremadamente importante), y debe ser trabajado como un ejemplo del concepto general de modelos de cómputo.

→ Pensamiento computacional

Este término se utiliza de modos muy diversos y con muy diferentes acepciones en diferentes contextos. Para esta propuesta consideramos la definición ofrecida por Alfred Aho (2011): “los procesos de pensamiento involucrados en la formulación de problemas de forma tal que sus soluciones puedan ser representadas como pasos computacionales y algoritmos”. La idea de algoritmo, como la entiende Aho, está íntimamente ligada a la noción de [modelo de cómputo](#), es decir que estos procesos de pensamiento involucran también la comprensión de los modelos de cómputo y en ocasiones la construcción de nuevos modelos de cómputo, cuando los existentes no son suficientes. Esta forma de entenderlo, tal como discuten Bonello y Schapachnik (2020), liga indisolublemente la noción de pensamiento computacional a la noción de computación y a las Ciencias de la Computación, tal como las presentamos en este documento. Es por todo esto que preferimos no utilizar el término pensamiento computacional y, en su lugar, usamos el término Ciencias de la Computación, que es más rico.

→ Prácticas computacionales

Se refiere al “saber hacer” de computación que se prioriza en la enseñanza obligatoria. Es en el marco de estas prácticas que los conceptos, incorporados en esta propuesta, se construyen, adquieren sentido y se organizan. Las prácticas computacionales, aun cuando se encuentran implicadas en la construcción de los saberes priorizados, se explicitan en forma separada para que el cuerpo docente pueda planificar situaciones de enseñanza a partir de ellas. Las cuatro prácticas que se destacan son:

- reconocer y desarrollar soluciones para problemas computacionales,
- crear y/o modificar [artefectos computacionales](#),
- desarrollar, utilizar y abrir [abstracciones](#),
- identificar y comunicar problemas de [dispositivos](#) y artefactos computacionales.

→ Representación simbólica de datos

En computación, los datos deben representarse en forma simbólica mediante una descripción codificada que pueda distinguirse de otras e identificar de forma unívoca al dato representado. Una forma típica de representación de datos es la representación binaria, que permite representar números utilizando cadenas de ceros y unos. Otra forma de representación es a través de estructuras de datos (tales como registros, listas, etc.) en lenguajes de programación, para modelar entidades y sus atributos. Estas estructuras adoptan diversas formas según los diferentes paradigmas, lenguajes de programación o aplicaciones que los utilizan. Al considerar las CC en sentido amplio, debemos tener en cuenta que la representación binaria es solamente un ejemplo de representación de información y debe ser trabajada como un ejemplo del concepto más general de representación de información. No debe confundirse el concepto de representación de datos con los usos que se les puede dar en otras disciplinas, como las que provienen del área de las Ciencias Sociales, por ejemplo.

→ [Representación de la información](#)

Ver [Representación simbólica de datos](#).

→ [Robótica](#)

La robótica hace referencia a por lo menos tres campos de estudio:

- A. La robótica industrial, que se dedica al diseño, confección y operación de maquinaria que realiza tareas repetitivas en contextos fabriles o similares. Se trata en general de dispositivos electromecánicos controlados digitalmente mediante un ciclo de control relativamente sencillo, más allá de que realicen tareas complejas.
- B. La robótica general, que se dedica al estudio, diseño y confección de entidades más bien autónomas, capaces de operar en ambientes no controlados y realizar planificaciones complejas para lograr su objetivo, es decir, un brazo mecánico que suelda partes en una fábrica tiene muy poco que ver con un robot de rescate que se mete entre escombros para buscar y auxiliar a víctimas.
- C. La robótica educativa, área donde se utilizan robots para reforzar aprendizajes de diversas disciplinas, por ejemplo, para calcular ángulos de giro en Matemática, temas de Física en el manejo de sensores, etc.

La robótica general cobra relevancia porque está presente en cada vez más áreas de la vida cotidiana. Esa comprensión requiere entender dos conceptos distintos: por un lado, cómo funcionan los sensores y actuadores que interactúan con el mundo físico y, por otro lado, cómo funcionan las computadoras que los controlan. Si consideramos que entre los objetivos de aprendizaje de la materia Física ya se encuentran presentes los conceptos de mecánica y del instrumental necesarios para entender el funcionamiento de un motor o la detección de un obstáculo por el rebote de una onda electromagnética, lo que resta incorporar es el entendimiento de los programas que controlan esos dispositivos (cubierto en los contenidos de programación). Es decir, teniendo conocimientos de Computación es posible comprender la forma en que funcionan los robots aun sin una formación específica en robótica porque se trata de un caso particular de una disciplina más general. En los años 90, se instaló la idea de que usar *software* de ofimática era el saber necesario que la escuela debía transmitir para que la ciudadanía pudiera participar del mundo digital. Esto se vuelve a expresar en la actualidad a través de la idea de que es imprescindible enseñar robótica. Como herramienta, el uso de

robots para el aprendizaje de otros temas (es decir, la robótica educativa) ha mostrado ser efectivo,⁹¹ y lo mismo puede decirse de su utilización con niños y niñas para iniciar el aprendizaje de la programación,⁹² ya que son elementos altamente motivadores. Es decir, valoramos su inclusión como un medio más que como un fin.

→ [Software libre](#)

Ver [Derechos de la ciudadanía asociados a los artefactos y dispositivos computacionales](#).

→ [Tecnología digital o computacional](#)

En este trabajo distinguimos el término computacional, que refiere al procesamiento de la información, del de digital, que se refiere a una forma de representar información (DiSessa, 2000, Jacob y Warschauer, 2018, Wing, 2006). El término digital hace referencia a una forma específica de representar la información basándose en cadenas de símbolos tomados de una cantidad finita de valores posibles (típicamente se utilizan dos, por lo que se habla de representación binaria) en lugar de las formas analógicas que utilizan un modelo continuo. Por ejemplo, los antiguos discos de vinilo o los casetes son formas analógicas de representar el sonido, mientras que las actuales formas utilizan representaciones basadas en la representación binaria. Así, al hablar de tecnología digital se está haciendo referencia a un estado particular de la tecnología de los dispositivos y las técnicas asociadas, a su representación y manipulación. El término *tecnología digital* se difundió de manera general y se utiliza muchas veces como sinónimo de *tecnología computacional*. Sin embargo, al pensar en la tecnología computacional hacemos referencia a los procesos lógicos y las formas de expresarlos de las cuales las formas digitales son solamente un ejemplo particular. En este trabajo elegimos el término tecnologías computacionales para poner el foco en los conceptos relevantes.

⁹¹ Ver, por ejemplo, Fabiane Barreto y Vavassori Benitti (2012), "[Exploring the educational potential of robotics in schools: A systematic review](#)", *Computers & Education*, Vol, 58 (3), abril: 978-988.

⁹² Por ejemplo, Martínez, Gómez y Benotti (2015).

Referencias bibliográficas

- Agencia de Noticias Ciencias de la Comunicación (ANCCOM), (2020). <http://anccom.sociales.uba.ar/>
- Aho, A. (2011). "Computation and Computational Thinking". *Ubiquity*, volumen 2011, enero. <http://ubiquity.acm.org/article.cfm?id=1922682>
- Argentina, Consejo Federal de Educación. Resolución 135/11. Núcleos de aprendizaje prioritarios de Educación Tecnológica, Anexo 3.
- Argentina, Consejo Federal de Educación. Resolución 343/2018. Núcleos de Aprendizaje Prioritarios para Educación Digital, Programación y Robótica.
- Argentina, Ministerio de Cultura y Educación de la Nación Consejo Federal de Cultura y Educación (1995). Contenidos Básicos Comunes para la Educación General Básica.
- Argentina, Unidad Fiscal Especializada en Ciberdelincuencia (2021). *Informe de gestión de la Unidad Fiscal Especializada en Ciberdelincuencia*. https://www.mpf.gob.ar/ufeci/files/2021/09/UFECI_informe-pandemia.pdf
- Association for Computing Machinery, Code.org, Computer Science Teachers Association, Cyber Innovation Center y National Math and Science Initiative (2016). *K-12 Computer Science Framework*.
- Barr, V. y Stephenson, C. (2011). "Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?", en *ACM Inroads*, 2(1), 48-54. https://www.researchgate.net/publication/247924673_Bringing_computational_thinking_to_K-12_what_is_Involved_and_what_is_the_role_of_the_computer_science_education_community
- BCS, Microsoft, Google y Intellect (2012). *Computer Science: A curriculum for Schools*. Reino Unido: Computing at School Working Group.
- Bell, T., Tymann, P., y Yehudai, A. (2018). "The big ideas in computer science for K-12 curricula". *Boletín de EATCS*, 1(124).
- Bernstein, B. (1973). *Class, Codes and Control*. Vol. I. Londres: Paladin
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). *Developing computational thinking in compulsory education - Implications for policy and practice*. JRC Science for Policy Report. doi: 10.2791/792158.
- Boix Mansilla, V. y Gardner, H. (2005). "¿Cuáles son las cualidades de la comprensión?", en *La enseñanza para la comprensión. Vinculación entre la investigación y la práctica*. Buenos Aires: Paidós.
- Bonello, M. B. y Schapachnik, F. (2020). "Diez preguntas frecuentes (y urgentes) sobre pensamiento computacional", en *Virtualidad, Educación y Ciencia*, 11(20): 1-11.
- Burke, Q., O'Byrne, W. I., y Kafai, Y. B. (2016). "Computational participation: Understanding coding as an extension of literacy instruction", en *Journal of adolescent & adult literacy*, 59(4), 371-375.
- Busaniche, B. (2007). "Alfabetización digital: las fronteras del aprendizaje y el control de la información", en Cabello, R. y Levis, D. (ed.), *Medios informáticos en la educación a principios del siglo XXI*. Buenos Aires: Prometeo.
- Centro de Estudios Legales y Sociales [CELS] (2022). "El ministerio de seguridad de la ciudad buscó información biométrica de 7 millones de personas de manera ilegal".

<https://www.cels.org.ar/web/2022/04/el-ministerio-de-seguridad-de-la-ciudad-busco-informacion-biometrica-de-7-millones-de-personas-de-manera-ilegal/>

Clavero, J. A. (2018). “Posverdad y exposición selectiva a *fake news*. Algunos ejemplos concretos de la Argentina”, en *Contratexto*, 29: 167-180.

Comisión Europea (2021). Ley Europea del Clima. Disponible en https://climate.ec.europa.eu/eu-action/european-climate-law_es

Comisión Europea (11 de diciembre de 2019). *El Pacto Verde Europeo* [Comunicación de la Comisión]. Consultado: 19 de agosto, 2023. Disponible en <https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=COM%3A2019%3A640%3AFIN>

Comisión Europea (23 de febrero de 2022). *Propuesta de Directiva del Parlamento Europeo y del Consejo sobre Diligencia Debida de las Empresas en Materia de Sostenibilidad y por la que se modifica la Directiva (UE) 2019/1937*. Consultado 19 de agosto, 2023. Disponible en <https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:52022PC0071&from=EN#footnoteref20>

Conference on Innovation and Technology in Computer Science Education [ITiCSE '15]: pp. 159-164.

Connell, R. (1997). “La justicia curricular”, en *Laboratorio de Políticas Públicas*, 6 (27), pp. 1–10.

CSTA (2017). *CSTA Standards*. Disponible en <https://csteachers.org/k12standards/>

Cucuzza, G. (16 de octubre de 2013). *La Informática debe ser una materia curricular*. Disponible en <https://www.lanacion.com.AR/opinion/la-informatica-debe-ser-una-materia-curricular-nid1629448>

Denning, P. J. (2004). “Great principles in computing curricula”, en *Boletín de ACM SIGCSE*, 36(1), pp. 336–341.

DiSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Mit Press.

Dussel, I. (2007). “La transmisión cultural asediada: los avatares de la cultura común en la escuela”, en *Propuesta Educativa*, 28, pp. 19-27.

Dussel, I. (2014). “¿Es el currículum escolar relevante en la cultura digital? Debates y desafíos sobre la autoridad cultural contemporánea”, en *Education Policy Analysis Archives/Archivos Analíticos de Políticas Educativas*, 22, pp. 1-22.

Dussel, I. (2020). “The Shifting Boundaries of School Subjects in Contemporary Curriculum Reforms”, en *Zeitschrift Fur Padagogik*, 66(5), pp. 666-689.

Echeveste, M. E., y Martínez, M. C. (2022). “El rol de los capitales digitales en escuelas técnicas de programación y las luchas estudiantiles por su lugar en el campo”, en *Revista de Sociología de la Educación RASE*, 14(3), pp. 307-324.

Echeveste, M. E., Gómez, M., Martínez, C. y Benotti, L. (2022). “La escuela y la brecha de género en la enseñanza de la computación”, en *Jornadas Argentinas de Didácticas de la Computación*. Fundación Sadosky.

Eglash, R., Bennett, A., Cooke, L., Babbitt, W., y Lachney, M. (2021). “Counter-hegemonic computing: Toward computer science education for value generation and emancipation”, en *ACM Transactions on Computing Education (TOCE)*, 21(4), 1-30.

Epstein, A. (rev., ed.). *The Intentional Teacher: Choosing the Best Strategies for Young Children's Learning*. Disponible en

https://www.ryders-hayes.co.uk/images/LFDAY1/Excerpt_Intentional_Teacher.pdf

Fattore, N. (2007). "Apuntes sobre la forma escolar 'tradicional' y sus desplazamientos", en Baquero, R., Diker, G., y Frigerio, G. (comp.). *Las formas de lo escolar*. Buenos Aires: Del Estante Editorial, pp. 13-32.

Ferreras, M. y Gay, A. (s.f.). *La educación tecnológica. Aportes para su implementación*. Argentina: Prociencia, MCyEN.

Furman, M. (2021). *Enseñar distinto. Guía para innovar sin perderse en el camino*. Buenos Aires: Siglo Veintiuno Editores.

Garzón, M. (2022). "La incorporación de las Ciencias de la Computación en el currículum: ¿la transversalidad como alternativa?", en *II Jornadas de Didáctica de las Ciencias de la Computación (JADICC)*. Universidad Nacional del Nordeste, Fundación Sadosky, 2020. Disponible en https://curriculum.program.ar/wp-content/uploads/2023/02/Garzon_Transversalidad_alternativa_JADICC2022.pdf

Gimeno Sacristán, J. (2010). "¿Qué significa el currículum?" (adelanto), en *Sinéctica*, (34), pp. 11-43. Disponible en: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-109X2010000100009&lng=es&tlng=es.

Goldar, M. G. (2021). "Tipos de obsolescencia y formas de combatirla desde el derecho privado", en *Análisis Jurídico-Político*, 3(6), pp. 231-252

Goode, J., y Margolis, J. (2011). "Exploring computer science: A case study of school reform", en *ACM Transactions on Computing Education [TOCE]*, 11(2), pp. 1-16.

Goode, J., y Chapman, G. (2015). *Exploring Computer Science*, versión 6.2.

Guerrero, A. B. (2001). "Estructuración de los contenidos del currículo. Funcionalidad y calidad", en *Cuestiones pedagógicas. Revista de Ciencias de la Educación*, 15.

Harari, Y. N. (2020). "The World After Coronavirus", en *Financial Times*, 20(03).

Hazzan, O., Lapidot, T. y Ragonis, N. (2014). *Guide to Teaching Computer Science*. Segunda edición. Londres: Springer-Verlag.

Isbell, C. L., Stein, L. A., Cutler, R., Forbes, J., Fraser, L., Impagliazzo, J. y Xu, Y. (2010). "(Re)defining computing curricula by (re)defining computing", en *ACM SIGCSE Bulletin*, 41(4), pp. 195-207. Disponible en <https://www.cc.gatech.edu/~isbell/papers/p195-isbell.pdf>

Jacob, S. R., y Warschauer, M. (2018). "Computational thinking and literacy", en *Journal of Computer Science Integration*, 1(1).

K-12 Computer Science Framework Steering Committee (2016). *K12 Computer Science Framework. Technical Report*. New York: Association for Computing Machinery.

Kafai, Y. B. (2016). "From computational thinking to computational participation in K-12 education", en *Communications of the ACM*, 59(8), pp. 26-27.

Ko, A. J. et al. (2020). "It is time for more critical CS education", en *Comm.ACM*, 63 (11), pp. 31-33.

Ko, A. J., Oleson, A., Ryan, N., Register, Y., Xie, B., Tari, M., Davidson, M., Druga, S. y Loksa, D. (2020). "It is time for more critical CS education", en *Communications of the ACM*, 63(11), pp. 31-33.

- Cucuzza, G. (16 de octubre de 2013). "La informática debe ser una materia curricular", en *La Nación*. Disponible en <https://www.lanacion.com.ar/opinion/la-informatica-debe-ser-una-materia-curricular-nid1629448>
- Lu, J. J. y Fletcher, G. H. (2009). "Thinking about computational thinking", en *Proceedings of the 40th ACM technical symposium on Computer science education*, pp. 260-264. Disponible en <https://www.win.tue.nl/~gfletche/papers-final/LuFletcherSIGCSE09.pdf>
- Maggio, M. (2018). "Habilidades del siglo XX: cuando el futuro es hoy", en *XIII Foro Latinoamericano de Educación (Santillana)*.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L. y Settle, A. (2014). "Computational thinking in K-9 education", en *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference*, pp. 1-29.
- Margolis, J., Estrella, R., Goode, J., Holme, J. J. y Nao, K. (2017). *Stuck in the shallow end: Education, race and computing*. MIT press.
- Margolis, J., Ryoo, J. J., Sandoval, C. D., Lee, C., Goode, J. y Chapman, G. (2012). "Beyond access: Broadening participation in high school computer science", en *ACM Inroads*, 3(4), pp. 72-78.
- Marino, V. (coord.), Sustas, S. E., Quartulli, D., Curcio, J. (2023). *Por qué estudiamos informática. Indagación sobre trayectorias universitarias: instituciones, estudiantes, género y trabajo*. Buenos Aires: Fundación Sadosky.
- Martínez, C. (2022). "Una mirada sobre la historia reciente de la computación en la escuela argentina", en Schapachnik, F. y Bonello, M. (2022), *Ciencias de la Computación en la escuela. Guía para enseñar mucho más que a programar*. Buenos Aires: Siglo XXI.
- Martínez, C.; Gómez, M. J. y Benotti, L. (2015). "A Comparison of Preschool and Elementary School Children Learning Computer Science Concepts through a Multilanguage Robot Programming Platform", en *ACM Conference on Innovation and Technology in Computer Science Education [ITiCSE '15]*, pp. 159-164.
- Martínez, M. C. y Echeveste, E. (2016). "El rol de las comunidades de aprendizaje en la construcción de una visión común para la enseñanza de computación en las escuelas", en *Revista Iberoamericana de Educación OEI/CAEU* (65), pp. 19-36.
- Martínez, C., E. P., Gómez, M., Borchardt, M., y Garzón, M. (2022). "Hacia un currículum emancipador de las Ciencias de la Computación", en *Revista Latinoamericana de Economía y Sociedad Digital*, 3. <https://doi.org/10.53857/LBUS5649>
- Pardo, J. (2021). "La imposición de integrar asignaturas en ámbitos", en *El diario de la Educación*, 9 de junio. https://eldiariodelaeducacion.com/2021/06/09/la-imposicion-de-integrar-asignaturas-en-ambitos/?fbclid=IwAR0PXXUv5P0INNagC-xKisAbmlDocQL2a5rRjfqjKdilb_4Qw81a882czA
- Roegiers, X. (2000). "Saberes, capacidades y competencias en la escuela: una búsqueda de sentido", en *Innovación educativa*, 10, pp. 105-122.
- Schapachnik, F. y Bonello, M. B. (2022). *Ciencias de la computación en la escuela. Guía para enseñar mucho más que a programar*. Buenos Aires: Siglo XXI editores.
- Simari, G. R. (2013). "Los fundamentos computacionales como parte de las ciencias básicas en las terminales de la disciplina Informática", en *VIII Congreso de Tecnología en Educación y Educación en*

Tecnología (TE&ET 2013), Santiago del Estero: Red de Universidades con Carreras en Informática (RedUNCI).

Simon, H. A. (1978: 2016). *Las ciencias de lo artificial*. España: Comares.

Society for Technology in Education [ISTE] (2016). *Standards for Students*. [En línea].

Storte, D., Webb, M., Bottino, R. M., et al. (2019). *Coding, programming and the changing curriculum for computing in schools*. UNESCO.

Szenkman, P. y Lotito, E. (2020). *Mujeres en STEM. Cómo romper con el círculo vicioso* [Documento 224]. Centro de Implementación de Políticas Públicas para la Equidad y el Crecimiento. Disponible en <https://www.cippec.org/wp-content/uploads/2020/11/224-DPP-PS-Mujeres-en-STEM-Szenkman-y-Lotito-noviembre-2020-1.pdf>

Terigi, F. (1999). *Conceptos para el análisis de políticas curriculares. Currículo: itinerario para aprehender un territorio*. Buenos Aires: Santillana.

Terigi, F. (2020). “Escolarización y pandemia: alteraciones, continuidades, desigualdades”, en *REVCOM*.

Tula, M. I. (2017). “Voto electrónico, controles, resguardos y garantía”, en Tullio, A., Pomares, J., y Leiras, M. (2017). *Reforma política en la Argentina: un debate democrático pendiente*. Rosario: UNR Editora.

Unesco (2019a), *I'd blush if I could: closing gender divides in digital skills through education*. Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. Disponible en: <https://en.unesco.org/ld-blush-if-i-could>

Unesco (2019b), *Descifrar el código: la educación de las niñas y las mujeres en ciencias, tecnología, ingeniería y matemáticas (STEM)*.

Unesco (2019c). *Coding, Programming and the Changing Curriculum for Computing in Schools. Report of UNESCO/IFIP TC3 Meeting at OCCE*, 27 de junio.

Unesco (2021). *Inteligencia artificial y educación. Guía para las personas a cargo de formular políticas*. París: Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura 7. <https://unesdoc.unesco.org/ark:/48223/pf0000379376.locale=es>

Vakil, S. (2018). “Ethics, identity, and political vision: Toward a justice-centered approach to equity in computer science education”, en *Harvard Educational Review*, 88(1), pp. 26-52.

Voogt, J., Fisser, P., Good, J., Mishra, P., y Yadav, A. (2015). “Computational thinking in compulsory education: Towards an agenda for research and practice”. en *Education and Information Technologies*, 20(4), pp. 715-728.

Wing, J. M. (2008). Computational Thinking and Thinking about Computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. Disponible en <https://doi.org/10.1098/rsta.2008.0118>

Zapata-Ros, M. (2018). *Pensamiento computacional. Una tercera competencia clave. El pensamiento computacional como una nueva alfabetización en las culturas digitales*. Murcia: Universidad de Murcia. Disponible en https://www.researchgate.net/profile/Miguel-Zapata-Ros/publication/322300195_Pensamiento_computacional_Una_tercera_competencia_clave/links/5a520dda0f7e9bbc10549211/Pensamiento-computacional-Una-tercera-competencia-clave.pdf

Propuestas curriculares analizadas para el desarrollo de esta propuesta

ACM Committee for Computing Education in Community Colleges (2017). *Computer Science Curricular Guidance for Associate-Degree Transfer Programs with Infused Cybersecurity*. Nueva York: Association for Computing Machinery.

Argentina, Consejo Federal de Educación. Resolución 135/11. [Núcleos de aprendizaje prioritarios de Educación Tecnológica](#), Anexo 3. [En línea].

Argentina, Consejo Federal de Educación. Resolución 343/2018. [Núcleos de Aprendizaje Prioritarios para Educación Digital, Programación y Robótica](#). [En línea].

Association for Computing Machinery, Code.org, Computer Science Teachers Association, Cyber Innovation Center y National Math and Science Initiative (2016). [K-12 Computer Science Framework](#). [En línea].

BCS, Microsoft, Google y Intellect (2012). *Computer Science: A curriculum for Schools*. Reino Unido: Computing at School Working Group.

CSTA (2017). [CSTA Standards](#). [En línea].

Goode, J., y Chapman, G. (2015). *Exploring Computer Science*, versión 6.2.

Hazzan, O., Lapidot, T. y Ragonis, N. (2014). *Guide to Teaching Computer Science*. Segunda edición. Londres: Springer-Verlag.

Ko, A. J. y otros (2020). "It is time for more critical CS education". *Comm.ACM*, 63 (11): 31–33.

K-12 Computer Science Framework Steering Committee (2016). *K12 Computer Science Framework*. Technical Report. Association for Computing Machinery. New York, NY, USA.

Society for Technology in Education [ISTE] (2016). [Standards for Students](#). [En línea]