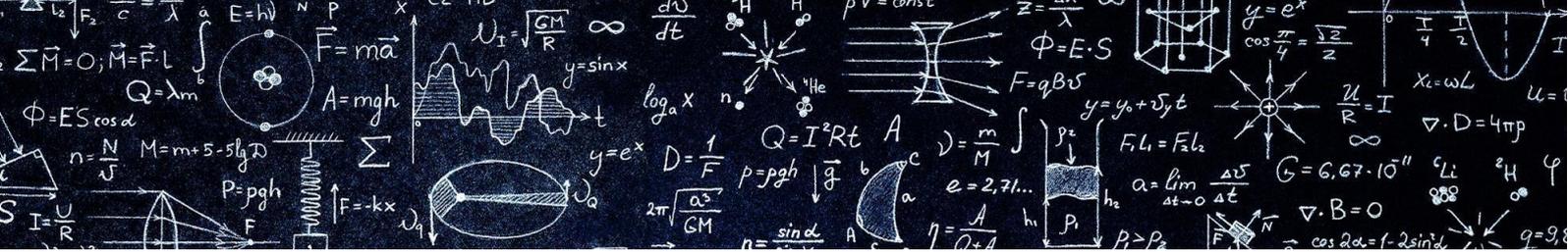


# Computational Thinking for Education Online

Número de proyecto: 2021-1-PL01-KA220-SCH-000024345

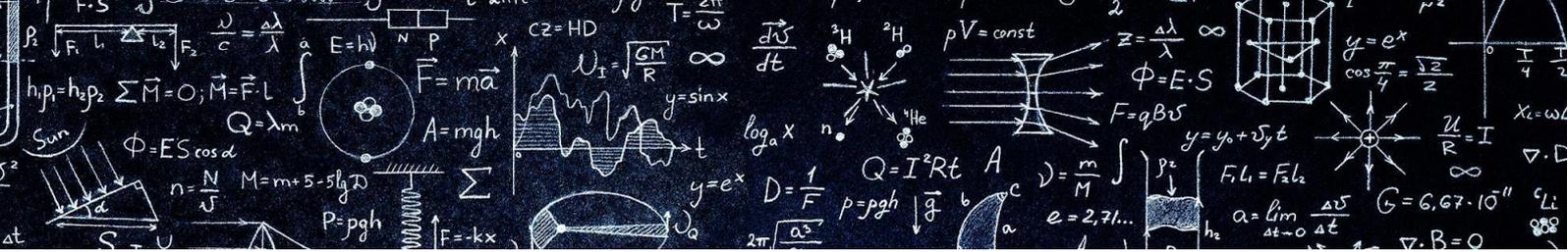




# Índice

<b>Capítulo 1: Introducción al Pensamiento Computacional (PC)</b>	<b>4</b>
1.1 Papel de la alfabetización digital en la educación moderna	4
1.2. ¿Qué es el Pensamiento Computacional?	8
1.3 Conceptos clave del Pensamiento Computacional	13
<b>Capítulo 2: Marcos de pensamiento computacional y educación en línea</b>	<b>23</b>
2.1 Introducción	23
2.2 Marcos de Pensamiento Computacional	24
2.2.1 Marco para estudiar y evaluar el desarrollo del pensamiento computacional por Brennan y Resnick (2012)	24
2.2.2 Marco Pedagógico de Pensamiento Computacional (MPPC) propuesto por Kotsopoulos et al. (2017)	26
2.3 Pensamiento Computacional en Educación Obligatoria: ¿Cómo incluir PC en los currículos escolares?	27
2.4 Modelo de Pensamiento Computacional para la Educación Online	33
2.4.1 Introducción	33
2.4.2 Funciones y responsabilidades de los docentes en la facilitación del aprendizaje en línea	35
2.4.3 Planificación e implementación de actividades para desarrollar habilidades de PC	38
2.5 Resumen	41
<b>Capítulo 3: Métodos de enseñanza en el pensamiento computacional</b>	<b>46</b>
3.1 Enfoques pedagógicos actuales y estrategias para enseñar pensamiento computacional	46
3.1.1 Modelo de Proceso de Diseño de Ingeniería en Educación PC	47
3.1.2 El Modelo de Aula Flipped Basado en 5E en la educación de PC	51
3.2 Enfoques y estrategias pedagógicos actuales para la enseñanza y el aprendizaje en línea	54
3.2.1 Herramientas de enseñanza en línea	56
3.2.2 PC en la educación en línea	69
3.3 Pensamiento computacional: Métodos de enseñanza	71

3.3.1 Formación en programación y computación matemática como método de enseñanza del Pensamiento Computacional	71
3.3.2 Aprendizaje experiencial: aprender haciendo en la educación PC	74
3.3.2.1 Aprendizaje basado en proyectos en las actividades de PC	77
3.3.2.2 Aprendizaje basado en problemas en las actividades de PC	80
3.3.2.3 Aprendizaje cooperativo en las actividades de PC	84
3.3.3 Los desafíos de la educación en PC y cómo superarlos	88
<b>Capítulo 4: Formación de Profesores en Pensamiento Computacional: Elementos de innovación e impacto esperado</b>	<b>100</b>
4.1. Desarrollo profesional eficaz del profesorado (TPD)	100
4.2 Formación de profesores en pensamiento computacional: Desafíos y Oportunidades	103
4.2.1 Desafíos para PC en la Educación	103
4.2.2 Beneficios de la formación del profesorado en PC	105
4.3 Aprendizaje Mediado Tecnológico (AMT) y Pensamiento Computacional (PC) para el Desarrollo Profesional del Educador	106
4.3.1 Programas de desarrollo profesional mediado por la tecnología (programas TTMPD)	107
4.3.2 AMT y PC	108
4.4 «Formación del entrenador»	110
4.4.1 ¿Por qué el pensamiento computacional?	110
4.4.2 Estrategias educativas para formadores	111
4.4.3 Herramientas y recursos para el autoestudio	114



Título de la publicación: Gamificación para la educación intercultural

Fecha y año de publicación: Julio de 2022

Atribución de la plantilla del libro: Sitio web de Docs and Slides, <https://docsandslides.com>

Más información sobre el proyecto en <http://edu-gamification.eu/>

## Organizaciones implicadas en el proyecto:

Krakowskie Centrum Zarządzania i Administracji Sp. A Zo.o.

Mediacreativa 2020 S.L.

2.º Peiramatiko Lykeio Athinas

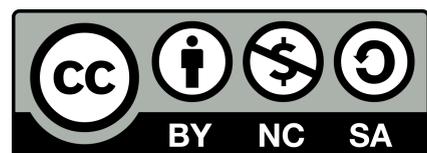
32 Escuela Secundaria de Idiomas "St. Kl. Ohridski"

## Autores y revisores:

- Anna Raşny, Krzysztof Karkowski, Dorota Karkowska
- Leire Monterrubio, Jaione Santos, Johanna Helmes
- Anastasia Mylona
- Veneta Velkova
- Ioannis Dimos, Athanasios Dimos



Cofinanciado por el programa Erasmus+ de la Unión Europea



La publicación está disponible bajo una licencia abierta Creative Commons Más información: [www.creativecommons.org/licenses/by-nc-sa/4.0/](http://www.creativecommons.org/licenses/by-nc-sa/4.0/)

*Esta publicación se ha producido con el apoyo financiero de la Comisión Europea en el marco del programa Erasmus+. Esta publicación solo refleja los puntos de vista de los autores, y la Comisión Europea no es responsable del contenido que contiene.*

## Capítulo 1: Introducción al Pensamiento Computacional (PC)

---

### 1.1 Papel de la alfabetización digital en la educación moderna

Las tecnologías modernas de la información y la comunicación ocupan un papel cada vez más importante en el mundo que nos rodea y participan directa o indirectamente en todas las áreas de nuestras vidas. Son muy atractivos porque, en cualquier contexto que se utilicen, ofrecen una nueva orientación en las situaciones, una nueva forma de pensar y una nueva visión del futuro. Se basan en la construcción de un nuevo sistema económico y social mundial — véase Dufva (2019); Yates y Rice (Ed), (2020); Bikalenko, Vekua, Telegina y Khabdaev (2021); Hilbert (2022); Nadoleanu (2022); Lynn (2022). La idea de utilizar la tecnología informática en la educación no es nueva. Durante años, todos los expertos en el campo han sugerido que cuando se utilizan de manera inteligente, pueden mejorar en gran medida los resultados de aprendizaje y mejorar la calidad del proceso.

Debido a la globalización y al aumento de la movilidad relacionada con el mercado laboral internacional, la tecnología es una prioridad para los sistemas educativos nacionales (Cuban, Kirkpatrick y Peck, 2001; Higgins y Packard, 2004). Ottestade y Quale (2009) encuentran una relación positiva entre la implementación de las TIC y los resultados del aprendizaje. Semenov (2005), Vrasidas y Glass (2005) y Roblyer (2006) proporcionan evidencia de que la integración de la tecnología en los currículos mejora el proceso de aprendizaje, aumenta los resultados de los estudiantes y proporciona mejores oportunidades para la enseñanza y el aprendizaje. Los autores afirman que, mediante la integración efectiva de las TIC en los planes de estudios, las instituciones educativas pueden proporcionar servicios y estrategias de mayor calidad.

El desarrollo dinámico de la tecnología en el siglo XXI y la aparición de una variedad de recursos de aprendizaje digital han ido acompañados de un replanteamiento de la naturaleza de la enseñanza y el aprendizaje. Conducen a una nueva comprensión del papel de la creación de materiales didácticos eficaces, a nuevos enfoques para enseñar a los profesores a integrar estos recursos, y a repensar la relación entre las actividades en las que los estudiantes están involucrados en la clase y su futura implementación. Todas estas tendencias dominantes provocan el desarrollo de una meta visión para la educación, mejorada significativamente mediante la integración y el uso de recursos digitales para la enseñanza y el aprendizaje, una educación que esté más estrechamente alineada con los requisitos para la futura alfabetización y realización de los jóvenes, una educación que pueda crear pensadores, capaces de trabajar en equipo y personalidades comunicativas que serán parte de una economía global cada vez más competitiva.

Para integrar las tecnologías de la información con éxito en el proceso educativo y garantizar su implementación en clase, es necesario un proceso de planificación y diseño cuidadosos: planificación de la formación y apoyo del profesorado, desarrollo de los nuevos materiales de aprendizaje digitales necesarios, coordinación de los elementos entre los diferentes niveles del sistema educativo, seguimiento y análisis periódicos.

Son importantes las actividades nacionales coordinadas sobre los factores relacionados con el éxito de la integración de las tecnologías digitales y el desarrollo del Pensamiento Computacional (PC) (explicado en la siguiente sección 1.2.) en la educación y la formación, así como la necesidad de un liderazgo integral en la innovación educativa. La enseñanza y el aprendizaje basados en la tecnología pueden conducir a muchos cambios en la escuela, lo que inevitablemente creará la demanda de nuevas formas de planificar el proceso de aprendizaje. Dudeney (2010) señaló que las

políticas nacionales de TIC pueden cumplir varias funciones clave. Proporcionan una justificación, un conjunto de objetivos y una visión de cómo funcionan los sistemas educativos si las TIC se integran en el proceso de enseñanza y aprendizaje, y son útiles para los estudiantes, los maestros, los padres y la población de un país.

La tecnología se puede utilizar para abrir el aula al mundo en general. Los profesores pueden beneficiarse de ella ampliando y profundizando sus habilidades profesionales y diversificando su estilo de enseñanza. Blackmore, Hardcastle, Bamblett y Owens (2003) argumentan que las nuevas tecnologías contribuyen tanto a la enseñanza como al aprendizaje mediante la creación de estudiantes independientes y motivados, alentando el uso de una variedad de métodos de enseñanza. Roblyer y Edwards (2000) creen que las TIC en la enseñanza ahorran tiempo y hacen que el aprendizaje sea más flexible. Utilizando tecnología informática, los profesores pueden simular el entorno de las características físicas de los objetos estudiados (Capron & Johnson, 2004).

Sin embargo, la integración de las TIC en el proceso educativo puede enfrentar una serie de barreras. Ertmer (1999) menciona dos tipos de barreras que enfrentan las instituciones educativas: barreras externas relacionadas con el acceso a un entorno TIC y la creación de un entorno de TIC y las capacidades de los profesores en el uso de la tecnología (capacidades digitales). El segundo tipo de obstáculos se definen como barreras internas y están relacionados con la calificación del profesor y sus actitudes y creencias. Cuando se superan las barreras externas, la decisión y la responsabilidad personal de si y cómo utilizar las TIC en el aula depende de las creencias, actitudes, motivación y competencias del docente.

Los beneficios de aplicar TIC y PC en la educación son muchos. Es importante reconocer que las personas aprenden mejor combinando imágenes con texto y sonido que utilizando cualquiera de los dos procesos

por sí solos, siempre que el diseño de los recursos de aprendizaje siga ciertos principios multimedia (Mayer & Moreno, 2003). Este conjunto de siete principios relacionados con multimedia y modalidad se basa en el trabajo de Richard Mayer, Roxanne Moreno y otros investigadores prominentes (Chan & Black, 2006; Ginns, 2005; Mayer, 2001; Mayer & Moreno, 2003).

1. **Principio multimedia:** La retención de los estudiantes se mejora a través de una combinación de palabras (verbal o texto) y visuales, en lugar de usar palabras solas, siempre que no introduzca redundancia de contenido.
2. **Principio de Contiguidad Espacial:** Los estudiantes aprenden mejor, cuando el texto y las imágenes se integran físicamente en lugar de estar separados.
3. **Principio de Contiguidad Temporal:** Los estudiantes aprenden mejor, cuando el texto y las imágenes correspondientes se sincronizan temporalmente en lugar de separarse en el tiempo.
4. **Principio de atención dividida:** Los estudiantes aprenden mejor cuando se excluyen palabras, imágenes y sonidos extraños en lugar de incluirlos.
5. **Principio demodalidad:** Los estudiantes aprenden mejor cuando el texto se presenta auditivamente como discurso en lugar de como texto en pantalla.
6. **Principio de diferencias individuales:** Los efectos de diseño de estos principios son más altos para los estudiantes de bajo conocimiento que para los estudiantes de alto conocimiento, y que son más altos para los estudiantes de alto nivel espacial que para los estudiantes de bajo nivel espacial.
7. **Principio de manipulación directa:** A medida que la complejidad de los materiales aumenta el impacto de la manipulación directa de los materiales de aprendizaje (animación, ritmo) en la transferencia también aumenta.

Como resultado, los estudiantes involucrados en el aprendizaje que viene con diseños multimodales de alta calidad superan, en promedio, a los estudiantes que aprenden utilizando enfoques tradicionales con modos individuales.

Vale la pena señalar que la digitalización del proceso educativo no solo se debe a la aparición de nuevos métodos y formas organizativas de educación, sino que conduce a un cambio significativo en el papel de los docentes y a un requisito de automejora y formación continua. Esta tendencia insiste en que los profesores tienen una serie de nuevas habilidades para gestionar el trabajo de clase aquí para mencionar entre ellos la capacidad de producir nuevos conocimientos, desarrollar nuevas formas de utilizar las tecnologías de la información y la comunicación para enriquecer el entorno de aprendizaje, y mantener una estrecha vigilancia en el desarrollo de la alfabetización digital de los estudiantes.

En resumen, el desarrollo profesional de los docentes es el punto de partida en el algoritmo de mejorar el sistema educativo e integrar las nuevas tecnologías en el proceso educativo.

La alfabetización digital debe desarrollarse y mejorarse estrechamente a través de la aplicación paralela del *aprendizaje basado en la indagación (ABI)*. A través de este último, los estudiantes se dedican a la investigación real y auténtica. *ABI* tiene como objetivo lograr un concepto más realista del conocimiento científico y proporcionar un entorno más motivador y orientado al aprendizaje.

*El aprendizaje basado en la investigación (ABI)* como sinónimo de *ABI* está ganando cada vez más apoyo (Polman, 1998). Para ser eficaz, el aprendizaje basado en la investigación debe incluir los pasos básicos para llevar a cabo la investigación y comprender la forma en que los científicos hacen descubrimientos científicos y, al mismo tiempo, desarrollar El PC con sus 4 pilares cardinales: descomposición, reconocimiento de

patrones, abstracción, pensamiento algoritmo (explicado más en la sección 1.3). La capacitación basada en la investigación se centra en la importancia de los procesos de «masterización» en las ciencias naturales, como la formulación de preguntas empíricamente estudiadas apoyadas por afirmaciones y evidencias" (Polman, 1998: 3). El uso efectivo de este método involucra a los estudiantes en la investigación independiente, los provoca a pensar profunda y científicamente darse cuenta de la conexión entre la evidencia y la teoría, y desarrollar PC. No es tanto el resultado del estudio lo que es significativo, sino el proceso del estudio y, por lo tanto, es importante proporcionar tiempo para la discusión y alentar a los estudiantes a presentar sus ideas (Watson, 2000).

## 1.2. ¿Qué es el Pensamiento Computacional?

El «pensamiento computacional» (PC) como concepto se ha vuelto popular en los últimos años; especialmente después de ser definido por Wing en 2006. Hasta hace poco, la computación era considerada una habilidad limitada poseída solo por científicos de computación, ingenieros, matemáticos y aquellos de disciplinas similares. Sin embargo, hoy en día casi todo el mundo, independientemente de la edad, se espera que tenga algunas habilidades básicas de computación en paralelo con los desarrollos en tecnología.

En un artículo seminal publicado en 2006, Jeanette Wing describió el pensamiento computacional (PC) como una forma de «resolver problemas, diseñar sistemas y comprender el comportamiento humano basándose en los conceptos fundamentales de la informática». Ella señaló que el pensamiento computacional involucra algunos conceptos familiares, como la descomposición de problemas, la representación de datos y el modelado, también ideas menos familiares, como la búsqueda binaria, la recursión y la paralelización. También argumentó: «El pensamiento computacional es una habilidad fundamental para todos, no solo para los informáticos».

PC es una concepción que ha ido creciendo durante los últimos años, utilizada por primera vez por Papert en 1996. En su artículo, Papert no definió exhaustivamente PC, pero en 2006, Jeannette Wing presentó PC y lo definió como una habilidad para todos, no solo para los informáticos. Esta descripción parece general y abstracta en términos de integración del PC en las clases y cómo observar la capacidad de PC de los académicos (Zhenrong, Wenming y Rongsheng, 2009). Empresas como Google y Microsoft apoyaron esta idea, y varios programas y proyectos han surgido para incorporar PC en diferentes currículos. Por otra parte, la Sociedad Internacional de Tecnología en la Educación (the International Society for Technology in Education (ISTE)) y la Asociación de Profesores de Ciencias de la Computación (Computer Science Teacher Association (CSTA)) publicaron una descripción funcional sobre PC; PC es un proceso basado en problemas que incluye (no limitado a) las siguientes características:

- Formular problemas de tal manera que nos permita utilizar una computadora y otras herramientas similares para ayudarnos a resolverlos;
- Organizar y analizar lógicamente los datos;
- Representar datos a través de abstracciones como modelos y simulaciones;
- Automatizar soluciones utilizando el pensamiento algorítmico (una secuencia de pasos ordenados);
- Identificar, analizar y aplicar posibles soluciones con el objetivo de lograr la combinación más eficaz y eficiente de medidas y recursos;
- Generalizar y transferir este proceso de resolución basado en problemas a una gran diversidad de problemas (CSTA e ISTE, 2011).

Además de esta definición, Mannila y sus colegas (2014) afirmaron que el PC podría ser un término que cubra un grupo de concepciones y procesos de pensamiento desde la computación que ayuden a formular problemas y sus soluciones en varias áreas temáticas. Del mismo modo, Riley y Hunt

(2014) abordaron las estrategias cognitivas del pensamiento, ya que «la mejor manera de caracterizar el Pensamiento Computacional es como la forma en que piensan los informáticos, la forma en que razonan» (p. 4).

Wing ve el PC como proceso y lógica en lugar de programación y codificación. Dos aspectos principales del PC basados en la perspectiva de Wing son:

- \_ pensar en múltiples niveles de abstracción durante los procesos de resolución de problemas;
- \_ comunicar ideas y conocimientos en términos computacionales durante la resolución de problemas y la investigación práctica.

Sysło y Kwiatkowska (2013) también enfatizan que el PC es un conjunto de habilidades de pensamiento que pueden no conducir a la programación de computadoras. Según ellos, el PC debería «enfocarse en los principios de la computación en lugar de en las habilidades de programación informática» (pág. 50). Al examinar las definiciones dentro de la literatura, la mayoría de ellas permanecen en la resolución de problemas, la comprensión de los problemas y la formulación de problemas (Wing, 2006; Zhenrong, Wenming y Rongsheng, 2009; Liu y He, 2014; Barr, Harrison y Conery, 2011). Para mejorar esta capacidad, palabras como algoritmo y precondition deben ser un barrio del vocabulario de todos (Zagami, 2013). Tras la categorización propuesta por Brennan y Resnick (2012), Lye y Koh (2014) sugirieron aspectos de PC sobre conceptos, prácticas y perspectivas. Con la dimensión de la terminología computacional, mencionaron concepciones utilizadas por expertos en TI como variables. Para el segundo aspecto, la aplicación de PC, tenían en mente las prácticas de resolución de problemas encontradas en los procesos de programación informática, como los bucles y la recursión. Como la última dimensión, a saber, las perspectivas computacionales, los autores mencionaron la antes mencionada (sección 1.1.) la superación y la

autorrealización de los estudiosos en términos del mundo tecnológico que los rodea, como la abstracción, el cuestionamiento y la depuración.

El pensamiento computacional (PC) es una habilidad fundamental que es equivalente a las habilidades de lectura, escritura y aritmética (National Research Council, 2005) PC implica resolución de problemas, diseño de sistemas y «comprensión de comportamientos humanos» (Wing, 2006). PC es también un proceso metacognitivo que consiste en sub-habilidades y disposiciones, que proporcionan a los estudiantes la capacidad de analizar patrones científicos y modelar fenómenos complejos (C. P. Dwyer, M. J. Hogan, e I. Stewart, 2013). Por lo tanto, el PC es el «tercer pilar» de la práctica científica Consejo Nacional de Investigación, 2005) y desempeña un papel crítico en la investigación científica y la resolución de problemas (PCAST), (2010).

El pensamiento computacional implica:

1. ***Pensamiento algorítmico***

El Pensamiento algorítmico es la capacidad de crear una secuencia ordenada de pasos con el objetivo de resolver un problema (más explicado en la siguiente sección).

2. ***Evaluación/Depuración***

Esta es la capacidad de verificar si un prototipo funciona según lo planeado, y si no, la capacidad de identificar lo que necesita mejorarse. Este es también el proceso por el que pasa un desarrollador para encontrar errores en el programa y solucionarlos.

3. ***Abstracción***

La abstracción es la capacidad de explicar un problema o una resolución eliminando detalles insignificantes.

Beneficios del desarrollo del Pensamiento Computacional en la escuela:

- ser capaz de conceptualizar una idea.

Un proceso para desarrollar habilidades de pensamiento computacional.

#### 4. **Planificación**

Los estudiantes deben pasar algún tiempo imaginando diferentes soluciones al problema, y luego hacer un plan en profundidad para ejecutar una entre sus ideas. Definirán los pasos por los que tendrán que pasar para llegar a la solución. Cuando identifican partes de la tarea que pueden haber visto antes, desarrollarán una habilidad llamada «Generalización».

El objetivo es hacer que los estudiantes

- ser capaz de hacer una lista de acciones para programar
- ser capaz de identificar partes de los programas existentes que podrían utilizar
- ser capaz de reutilizar partes de los programas

A continuación, los estudiantes tienen la tarea de crear la versión definitiva de su solución.

Cuando los estudiantes codifican sus ideas, desarrollan sus habilidades de Pensamiento Algorítmico.

#### 5. **Modificación**

Los estudiantes evalúan sus soluciones de acuerdo a si su programa y modelo cumplen o no los criterios de éxito. Usando sus habilidades de evaluación, identificarán si necesitan cambiar, arreglar, depurar o mejorar una parte de su programa.

El propósito es hacer que los estudiantes:

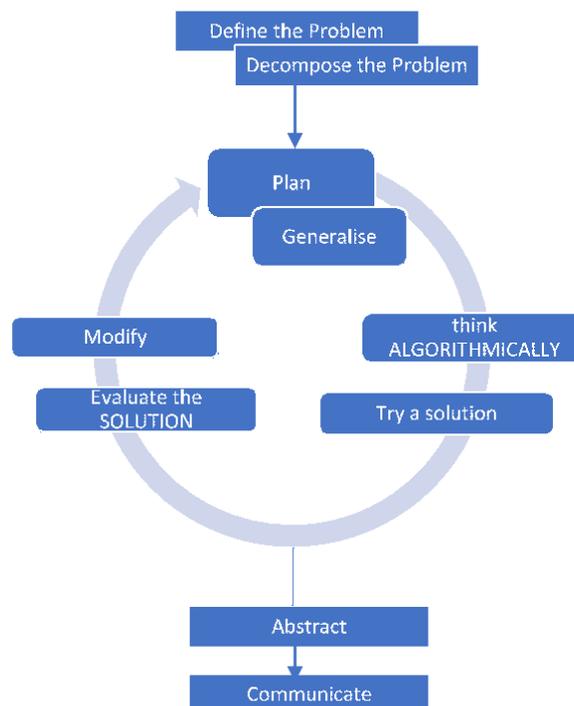
- realización de iteraciones de su programa
- solucionar problemas en su programa
- capaz de evaluar si la solución está vinculada al problema

## 6. **Comunicación**

Cada estudiante presentará la versión final de su solución a la clase, explicando cómo su solución cumple con los criterios de éxito. Al explicar su solución con el nivel adecuado de detalle, van a desarrollar sus habilidades de abstracción y comunicación.

El objetivo es hacer que los estudiantes:

- explicar la parte más importante de su solución
- dar suficiente detalle para mejorar la comprensión
- aclarar cómo su solución cumple con los criterios de éxito



### 1.3 Conceptos clave del Pensamiento Computacional

El pensamiento computacional se puede definir más como un conjunto de habilidades para la resolución de problemas basados en técnicas informáticas. El pensamiento computacional se requiere en todas partes y va a ser una clave para el éxito en la mayoría de las carreras, no solo para un científico, excepto para muchos profesionales, como médicos, abogados, maestros o agricultores. Para resolver un problema, es una buena idea hacer un plan utilizando algunas de las técnicas de informática como: descomponer un problema complejo en partes más pequeñas que son más controlables y más fáciles de entender o resolver: la descomposición; identificación de similitudes entre y dentro de los problemas y otras prácticas — reconocimiento de patrones; centrarse solo en información importante y extraer diferencias específicas para hacer que una solución funcione para muchos problemas diferentes: la abstracción; desarrollar una solución paso a paso al problema: algoritmos. Todos, sin importar su área de especialización, tarea o edad, a menudo emplean este plan. Es importante practicar y desarrollar estas técnicas muy temprano. En los últimos años, hemos de constatar la proliferación de diversos proyectos con el objetivo preciso de fomentar el estudio del pensamiento computacional. Los proyectos de masificación del pensamiento computacional y la codificación están comenzando a implementarse en varios sistemas educativos.

El pensamiento computacional transforma a los educadores en innovadores, para encontrar formas de resolver un problema, organizar y planificar la resolución de una tarea. PC nos enseña y nos da el coraje, los métodos y técnicas para resolver problemas complejos.

En los últimos años, parte de la investigación relacionada con la formación y desarrollo de las competencias necesarias de los docentes para la aplicación de las TIC se basa en el marco conceptual del Conocimiento Tecnológico Pedagógico y de Contenido TPACK (TPACK). Koehler y Mishra (Koehler & Mishra 2005, 2007; Mishra & Koehler, 2006) lo propuso como una extensión del modelo Pedagogical Content Knowledge desarrollado por Shulman (Shulman, 1986).

Mishra y Koehler proponen este marco conceptual para describir el complejo sistema de conocimientos, habilidades y competencias, que debe tener un profesor que aplique la tecnología de la información moderna en su trabajo. A través de ella, esbozan el complejo «sistema de conexiones, interacciones, determinación mutua y limitaciones entre el contenido de la asignatura, la pedagogía y la tecnología». En este modelo, el conocimiento de la materia, la pedagogía y la tecnología es esencial para la formación de un buen maestro. Por mucho que se consideren como tres partes separadas del conocimiento del maestro, este modelo enfatiza aún más la compleja interacción entre ellos (Mishra y Koehler, 2006).

El pensamiento computacional tiene cuatro pilares estrechamente interrelacionados que generalmente se aplican sistemáticamente:

**La descomposición** se refiere a la operación de descomponer un sistema complejo en componentes más pequeños que son más fáciles de comprender, administrar, explorar y comprender. Cada problema más pequeño se resuelve en su propio tiempo y algunos pequeños detalles a su alrededor podrían tenerse en cuenta fácilmente. La descomposición nos da la oportunidad de examinar un problema/situación, etc. desde un amplio alcance de dimensiones y ángulos, y de comprender cómo funciona desde su núcleo más interno. En el proceso de descomposición de un problema más grande y más intrincado, a menudo detectamos/reconocemos patrones.

**El reconocimiento** de patrones (PR) podría describirse como la segunda característica fundamental de la informática. Los patrones representan similitudes o características que son comunes para los problemas/contenido de los procesos del sistema (texto, imágenes, sonidos, etc.). La naturaleza de los patrones podría describirse como secuencial o basada en el ritmo. Por lo tanto, el reconocimiento de patrones incluye la detección de estas similitudes/patrones entre los problemas descompuestos/segmentados y nos permiten resolver el problema más grande con mayor facilidad. Hay dos operaciones de relaciones públicas importantes:

- *explorativo* - utilizado para reconocer puntos comunes en los datos
- *descriptivo* – utilizado para categorizar las características comunes de una manera determinada (<http://theapp solutions.com>).

Además, la lista de próximas conferencias de la Asociación Internacional para el Reconocimiento de Patrones (the International Association for Pattern Recognition (IAPR)) suscita un gran interés: el alcance temático es amplio y abarca desde la interacción informática humana hasta el monitoreo ambiental (véase [iapr.org](http://iapr.org)).

**La abstracción** es una habilidad consciente y cognitiva muy alta que los seres humanos poseen predominantemente. Nos brinda la oportunidad de filtrar el flujo de datos híbridos y borrar la cantidad irrelevante de información (generalmente los detalles) para enfatizar la que queremos o necesitamos para el exitoso proceso de resolución de problemas. Esta operación selectiva resulta en crear una representación pura, imagen mental, idea de en qué consiste nuestro problema y crear un modelo o un algoritmo a seguir.

**El pensamiento algorítmico** consiste en construir una serie de pasos concretos, reglas, direcciones, instrucciones para resolver un problema. AT podría definirse además como la elaboración de una estrategia bien

ordenada, un plan de acción eficiente y fácil de seguir para hacer frente a algunas tareas complicadas. Es necesario establecer un punto de partida, un punto de finalización, un espectro de pasos a seguir en el medio. Los algoritmos son la herramienta líder para la programación de computadoras y la identificación de patrones.

### **Competencia de los docentes en TIC según el marco conceptual de la UNESCO**

El marco propuesto por un equipo de la UNESCO (UNESCO, 2008.a) hace hincapié en que no es suficiente que un profesor moderno sea científicamente alfabetizado y capaz de formar habilidades y hábitos científicos relevantes en sus estudiantes. El profesor moderno debe ser capaz de ayudar a los estudiantes a utilizar las TIC para trabajar juntos con éxito, para hacer frente a los problemas emergentes, para aprender habilidades de aprendizaje y, en última instancia, para hacer que los estudiantes, ciudadanos y empleados contribuyentes, responsables y competitivos.

Las normas desarrolladas en este marco están relacionadas con los siguientes aspectos del trabajo del profesor — módulos: comprender el papel de las TIC en la educación; contenido de aprendizaje y evaluación; pedagogía (habilidades didácticas); Competencias en materia de TIC; organización y gestión de la educación; Desarrollo profesional.

La idea principal de los autores de este sistema de estandarización se basa en tres enfoques de aprendizaje adoptados por ellos, que están relacionados con las respectivas etapas de desarrollo profesional de los profesores que aprenden a trabajar con las TIC. El primer enfoque — «Alfabetización Tecnológica» — requiere que los profesores sean capaces de ayudar a los estudiantes a utilizar las TIC para mejorar el aprendizaje. El segundo — «Profundidad del Conocimiento» — requiere que los profesores ayuden a los estudiantes a dominar más plenamente el plan de

estudios en la materia y aplicar los conocimientos adquiridos para resolver problemas complejos que ocurren en el mundo real. El tercer enfoque, «Crear conocimiento», requiere las habilidades de los profesores para ayudar a los estudiantes, futuros ciudadanos y trabajadores a producir (crear, adquirir) nuevos conocimientos necesarios para el desarrollo armonioso y la prosperidad de la sociedad.

El pensamiento computacional a menudo se ve como explícitamente relacionado con la programación, la codificación y la CC (Ciencias de la Computación), especialmente después de la publicación del documento seminal de Wing (Wing 2006). Los esfuerzos para desarrollar el PC de los estudiantes tienden a desarrollar y utilizar actividades, herramientas o plataformas asociadas con los conocimientos y habilidades de programación en intervenciones y programas educativos (Barth-Cohen et al. 2018; Hsu et al. 2018). Como ejemplo, Barth-Cohen et al. (2018) investigó cómo los estudiantes de quinto grado interpretan y navegan la información al participar en diversas actividades de codificación y resolución de problemas durante un entorno de programación. La escuela adoptó y utilizó un plan de estudios de robótica en su estudio, tenía software instalado en computadoras portátiles proporcionadas por la escuela para niños universitarios y tenía un robot físico para uso instructivo. El desarrollo de PC de los estudiantes fue examinado con un enfoque en su desempeño en la formulación y resolución de problemas durante este entorno de programación robótica (Ji Shen, Guanhua Chen, Lauren Barth-Cohen, Shiyang Jiang & Moataz Eltoukhy (2022). Conectando el pensamiento computacional en el razonamiento cotidiano y la programación para estudiantes de primaria, *Journal of Research on Technology in Education*, 54:2, 205-225).

Sin duda, la digitalización tiene un impacto cada vez mayor en la educación escolar. Últimamente, en contextos políticos, esto se ha descrito constantemente como el término «educación digital». Si bien

este término se asocia con frecuencia con el uso de la tecnología, también hay enfoques que se concentran adicionalmente en el pensamiento computacional. PC describe patrones de hacer frente a un problema como lo haría un científico informático (Wing, 2006). Aunque no hay una definición completa, existe un acuerdo mutuo en que incluye un conjunto de habilidades para pensar y lidiar con los problemas (Kalelioglu, Gülbahar, & Kukul, 2016). Se trata de aplicar la experiencia, como la abstracción o la avería, para que la resolución pueda ser llevada a cabo efectivamente por una computadora. Los enfoques originales para incluir el PC en la enseñanza secundaria se relacionan principalmente con la educación científica, por ejemplo, mediante la enseñanza de la ciencia a través de la simulación y la modelización (Basu et al, 2013). Pero los enfoques y logros de las humanidades digitales muestran que la enseñanza en otras áreas también podría estar fuertemente influenciada por la digitalización. En consecuencia, algunas fuentes consideran la incorporación de PC en las diferentes áreas temáticas (Barr & Stephenson, 2011, Kale et al., 2018). El ajuste de curvas o el análisis verbal de las oraciones son algunos ejemplos (Barr & Stephenson, 2011). El modelo de Aprendizaje Computacional (AC) enfatiza el importante papel que «la computadora (y posiblemente su abstracción) puede desempeñar para mejorar el proceso de aprendizaje y mejorar el rendimiento de los estudiantes en el campo de STEM y otros cursos» (Cooper, Pérez & Rainey, 2010). El modelo combina teorías de aprendizaje y la capacidad de la computadora para lidiar con la complejidad y visualizar los resultados de una manera adecuada para mejorar la comprensión, así como el aprendizaje. Si bien este concepto implica el uso de computadoras, por ejemplo, para simulaciones y modelado, los autores enfatizan que el modelo excluye explícitamente los usos no cognitivos de tecnologías como clickers o blogs. Los maestros pueden involucrar a sus estudiantes en PC y AC debido a habilidades y conocimientos en ciencias de la computación. Algunas universidades comenzaron a aplicar cursos de CC

para muchos grupos de estudiantes que no son de CC, mientras que hay una gran cantidad de formación de profesores previos al servicio que se concentran principalmente en desarrollar y mejorar las habilidades de tecnología de la información y la comunicación (TIC) (Goktas, Yildirim, & Yildirim, 2009).

Independientemente de la importancia del Pensamiento Computacional en todas las áreas temáticas a lo largo de la educación, los enfoques existentes que incluyen explícitamente las habilidades y competencias relacionadas con el PC en la formación de maestros que no son de EC siguen siendo poco comunes. Los cursos de CC existentes para profesores no-CC tienden a concentrarse, por ejemplo, en los fenómenos cotidianos que analizan y explican desde el punto de vista de las ciencias de la computación (Müller, Frommer, & Humbert, 2013), en las concepciones algorítmicas (Yadav et al., 2011), o en enfocar el pensamiento computacional en el contexto de la alfabetización de la información y los medios de comunicación (Dengel & Heuer, 2018).

Los conocimientos y habilidades en ciencias de la computación son necesarios para conocer la transformación digital y los avances dentro de las áreas temáticas. Además, la transformación digital también lleva a que nuevos temas sean relevantes para ser discutidos en clase (cf. Brinda & Diethelm, 2017). Debido a su historia de educación, muchos futuros maestros todavía no tienen una base correcta en computación. Por lo tanto, un curso debe ser apoyado por las ideas básicas de la informática que subyacen a la transformación digital y destacan los impactos en la sociedad. Para ser adecuado para futuros profesores de todas las disciplinas, un curso correspondiente también debe tener en cuenta los diversos niveles de conocimiento previo.

## **REFERENCIAS**

Дамянова, А. (2005). Конструктивизмът - новата образователна парадигма? Български език и литература (ел. версия), (5).

<https://litenet.bg/publish3/adamianova/konstruktivizmyt.htm>

Тафрова-Григорова, А. Кирова, М. & Бояджиева, Е. (2012). Учителите по природни науки – за конструктивистката учебна среда в българското училище. *Химия*, 21, 375-388.

Тошев, Б. В. (2007а). Ранното обучение по природни науки-интегрален или модулен подход. *Bulgarian Journal of Science & Educational Policy*, 1, 51-56.

Тошев, Б. В. (2009а). Метод на проектите в образованието. *Химия*, 18, 243-249.

Тошев, Б. В. (2012). Конструктивизъм – теория и практика. *Химия*, 21, 463-468.

Холенбек, Д. Кирова, М. Бояджиева, Е. & Тафрова-Григорова, А. (2009). Ученици и учители за настоящата и предпочитана учебна среда – резултати от едно проучване. *Химия*, 18, 349-369

Цанова, Н. & Райчева, Н. (2012). Методика на обучението по биология. Теория и практика. София: Пенсофт.

National Research Council (NRC), *How Students Learn: History, Science, and Mathematics in the Classroom*. Washington, DC: National Academy Press, 2005.

J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33-36, 2006.

C. P. Dwyer, M. J. Hogan, and I. Stewart, "An integrated critical thinking framework for the 21st century," *Thinking Skills and Creativity*, vol.12, pp. 43-52, 2014.

President's Council of Advisors on Science and Technology (PCAST). (2010). *Prepare and Inspire: K-12 Education in Science, Technology, Engineering, and Math (STEM) for America's Future*. Washington, DC: White House Office of Science and Technology Policy (OSTP) 2010.

Barr, D., Harrison, J., & Conery, L. (2011). Computational Thinking: A Digital Age Skill for Everyone. *Learning & Leading with Technology*, 38(6), 20-23.

Barr, V., & Stephenson, C. (2011). Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? *ACM Inroads*, 2(1), 48-54.

Barth-Cohen, L.A., Jiang, S., Shen, J., Chen, G., & Eltoukhy, M. (2018). Interpreting and navigating multiple representations as central to computational thinking in a robotics-programming environment. *Journal for STEM Education Research*, 1 (1-2), 119-147.

Brinda T., & Diethelm, I. (2017). Education in the Digital Networked World. *Proceedings of WCCE 2017*. Cham: Springer, 653-657.

Cooper, S., Pérez, L., & Rainey, D. (2010). K-12 Computational Learning. *Communications of the ACM*, 53(11), 27-29.

Goktas, Y., Yildirim, S., & Yildirim, Z. (2009). Main Barriers and Possible Enablers of ICTs Integration into Pre-service Teacher Education Programs. *Journal of Educational Technology & Society*, 12(1), 193-204.

Ji Shen, Guanhua Chen, Lauren Barth-Cohen, Shiyang Jiang & Moataz Eltoukhy (2022) Connecting computational thinking in everyday reasoning and programming for elementary school students, *Journal of Research on Technology in Education*, 54:2, 205-225

Kale, U., Akcaoglu, M., Cullen, T., Goh, D., Devine, L., Calvert, N., & Grise, K. (2018). Computational What? Relating Computational Thinking to Teaching. *TechTrends*, 62(6), 574-584.

Kalelioglu, F., Gülbahar, Y., & Kukul, V. (2016). A Framework for Computational Thinking based on a Systematic Research Review. *Baltic Journal of Modern Computing*, 4(3), 583.

Müller, D., Frommer, A., & Humbert, L. (2013). Informatik im Alltag-Durchblicken statt Rumklicken [CS in Everyday Life - Understanding Instead of Clicking Around]. *Proceedings of HDI '13*, 98-104.

Wing, J. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33-35.

Mayer, R. (2001). *Multi-media learning*. Cambridge, UK: Cambridge University Press.

Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. In R. Bruning, C. A. Horn, & L. M. PytlikZillig (Eds.), *Web-based learning: What do we know? Where do we go?* (pp. 23–44). Greenwich, CT: Information Age Publishing.

Cuban, L., Kirkpatrick, H., & Peck, C. (2001). High access and low use of technologies in high school classrooms explaining an apparent paradox.

Blackmore, J., Hardcastle, L., Bamblett, E. & Owens, J. (2003). *Effective use of information and communication technology (ICT) to enhance learning for disadvantaged school students*. Canberra, ACT: Department of Education, Science and Training.

Davis, F.D. (1993). User acceptance of information technology: system characteristics, user perception and behavioral impacts. *Int. J. Man-Machine Studies*, 38, 475-487

Davis, F. D., Bagozzi, R. P. & Warshaw, P. R. (1989). User acceptance of computer technology: A comparison of two theoretical models. *Management Science*, 35, 982–1003.

Dirckinck-Holmfeld, L., Hodgson, V., Jones, C, de Laat, M., McConnell D. & Ryberg, T. (2010). Teacher Use of ICT: Challenges and Opportunities. In: Dirckinck-Holmfeld, L., Hodgson, V., Jones, C., de Laat, M., McConnell, D. &

Ryberg, T. (Eds). *Proceedings of the seventh International Conference on Networked Learning*. (pp 439-445). Adlborg Universitet.

Donnelly, D., McGarr, O. & O'Reilly, J. (2011). A framework for teachers' integration of ICT into their classroom practice. *Computers & Education*, 57(2),1469-1483.

Ertmer, B. (1999). Addressing first- and second-order barriers to change: Strategies for technology integration. *Educational Technology, Research and Development*, 47(4), 47-61.

Kumar, R. & Gupta V.K. (2009): An introduction to cognitive Constructivism in Education. *Journal of Indian Education*. Nov.2009. New Delhi: NCERT.

Mishra, P. & Koehler, M.J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record*, 108,1017-1054.

Polman, J.L. (1998). *Why train "little scientists": The purposes and practices of science education in today's democracy*. Paper presented at the Annual Meeting of the American Educational Research Association, San Diego, CA.

Roblyer, M. & Edwards, J. (2000). *Integrating educational technology into teaching* (second Ed.). Upper Saddle River, NJ: Prentice Hall.

Roblyer, M. (2006). *Integrating educational technology into teaching* (fourth Ed.). Upper Saddle River, NJ: Pearson Education.

Semenov, A. (2005). *Information and communication technologies in schools: A handbook for teachers, or how ICT can create new, open learning environments*. Paris, France: United Nations Educational, Scientific and Cultural Organisation.

Shulman, L. (1986). Those who understand: Knowledge growth in teaching. *Educational Researcher*, 15, 4-14.

Singh, S. & Yadivashi, S. (2015). Constructivism in science classroom: Why and how. *International Journal of Scientific and Research Publications*, 5(3), 1-5.

UNESCO. (2008). ICT Competency Standards for Teachers. Policy Framework. <http://cst.unesco-ci.org/sites/projects/cst/The%20Standards/ICT-CST-Policy%20Framework.pdf>

UNESCO. (2008a). ICT Competency Standards for Teachers. Paris: United Nations Educational, Scientific and Cultural Organization.

UNESCO. (2008b). ICT Competency Standards for Teachers. Implementation Guidelines. <http://cst.unescoci.org/sites/projects/cst/The%20Standards/ICT-CS Implementationpercentage20Guidelines.pdf>

Watson, R. (2000). The role of practical work. In: Monk, M & Osborne, J (Eds.). *Good practice in science teaching – What research has to say*. (57-71). Buckingham: Open University Press.

## Capítulo 2: Marcos de pensamiento computacional y educación en línea

---

### 2.1 Introducción

El pensamiento computacional (PC) es parte de un término más amplio de alfabetización digital, y se considera una *habilidad importante del siglo XXI*. El mayor beneficio del PC es cómo permite la resolución de problemas en el mundo real, ayuda a dividir los grandes problemas en partes pequeñas y ofrece soluciones en una variedad de campos. Con conocimientos de PC, los estudiantes pueden aprender a resolver problemas matemáticos o escribir informes de libros (Smith, 2014).

Hay muchos marcos teóricos relacionados con el PC, los presentados en este capítulo proporcionan conocimientos útiles para un profesor cuya tarea es diseñar lecciones en línea para desarrollar habilidades de PC en los estudiantes.

El siguiente material le dará al lector una mejor comprensión de lo que es el PC y qué actividades se pueden hacer para desarrollar PC en los estudiantes. La pandemia de COVID-19 ha acelerado la transición al modo de enseñanza en línea, por lo que en este manual nos centraremos en el desarrollo del PC en este contexto.

Se presentarán los marcos teóricos de PC. Entre ellos, el marco de Brennan y Resnick (2012) y el Marco Pedagógico del Pensamiento Computacional de Kotsopoulos et al. (2017). Luego, se explicará cómo integrar el PC en todo el plan de estudios.

## **2.2 Marcos de Pensamiento Computacional**

### **2.2.1 Marco para estudiar y evaluar el desarrollo del pensamiento computacional por Brennan y Resnick (2012)**

Brennan y Resnick (2012) desarrollaron un marco que puede ayudar a comprender mejor el PC. Estaban interesados en presentar cómo las actividades basadas en proyectos apoyan el desarrollo del PC en los jóvenes, y en los últimos años desarrollaron un marco para resumir sus conclusiones.

Los autores basaron su teoría en actividades realizadas con la herramienta PC — Scratch — un entorno de programación que permite a los jóvenes crear sus propias historias interactivas, juegos y simulaciones, y luego compartir esas creaciones en una comunidad en línea con otros jóvenes programadores de todo el mundo.

En su modelo, Brennan y Resnick (2012) distinguen:

- **Conceptos de pensamiento computacional:** Los conceptos que los jóvenes encuentran a medida que se programan.
- **Estrategias y prácticas de pensamiento computacional:** las estrategias y prácticas que los jóvenes se adaptan para desarrollar medios interactivos.

En cuanto a los conceptos computacionales, distinguimos lo siguiente:

- **Secuencias:** una actividad o tarea en particular se expresa como una serie de pasos o instrucciones individuales que pueden ser ejecutados por una computadora.
- **Bucles:** un mecanismo para ejecutar la misma secuencia varias veces.
- **Paralelismo:** secuencias de instrucciones que ocurren al mismo tiempo.
- **Eventos:** una cosa hace que suceda otra cosa.
- **Condiciones:** toma de decisiones basada en ciertas condiciones, lo que apoya la expresión de múltiples resultados.
- **Operadores:** permita a los programadores realizar manipulaciones numéricas y de cadenas, incluyendo adición, resta, multiplicación, división, etc.
- **Datos:** implica almacenar, recuperar y actualizar valores.

Brennan y Resnick (2012) nombran las siguientes prácticas computacionales:

- **Pruebas y depuración:** desarrollado a través de ensayo y error, y transferido de otras actividades, o apoyo de otros conocedores.
- **Reutilización y remezcla:** en otras palabras, se basa en el trabajo de otras personas.
- **Abstracción y modularización:** es crear algo grande al juntar colecciones de piezas más pequeñas, es una práctica importante para todo diseño y resolución de problemas.

Los jóvenes están rodeados de medios interactivos. Los sitios web que utilizan para ser creados requieren habilidades de Pensamiento Computacional de

sus creadores. El objetivo de enseñar a los niños PC debe ser cambiar su perspectiva: de un consumidor a una persona que piensa «Puedo crear algo así» o «Sé cómo se hizo esto».

La creatividad y el aprendizaje son parte del proceso computacional. Son prácticas sociales que enriquecen el proceso y deben estructurarse a través de interacciones en línea y cara a cara. Al crear con otros, los jóvenes son capaces de hacer más de lo que podrían tener por su cuenta.

El marco de Brennan y Resnick (2012) ofrece una comprensión de las capacidades de PC e ilustra las vías para la adquisición de conocimiento que se abre para los estudiantes. Los autores destacan la importancia de evaluar las habilidades adquiridas durante las actividades de aprendizaje. El marco teórico es un mapa que nos muestra el camino, pero la etapa clave de trabajar con los estudiantes debe ser determinar si adquirieron las habilidades necesarias después de que el aprendizaje haya terminado.

Para determinar si la actividad de aprendizaje fue un éxito, los profesores deben evaluar a los estudiantes buscando respuestas a las siguientes preguntas:

- ¿Pueden poner en práctica el conocimiento que adquirieron durante la lección?
- ¿Pueden remezclar o reutilizar conceptos empleados por otros?
- ¿Son capaces de evaluar sus códigos y los códigos de sus pares?
- ¿Saben cómo depurar códigos problemáticos?

Brennan y Resnick's Framework capturaron el qué, cómo y por qué en su marco de PC, sin embargo, no abordaron completamente la enseñanza real del PC. Es por eso que el siguiente marco llenará el vacío en la comprensión del aspecto pedagógico del desarrollo del Pensamiento Computacional con los estudiantes.

### 2.2.2 Marco Pedagógico de Pensamiento Computacional (MPPC) propuesto por Kotsopoulos et al. (2017)

Se han propuesto numerosos marcos para el PC, pero el marco que describe la pedagogía ha sido un trabajo único y pionero de Kotsopoulos et al. (2017). Los autores proponen el Marco Pedagógico CT (MPPC), que incluye cuatro experiencias pedagógicas:

- **Las experiencias desconectadas** se centran en actividades implementadas sin el uso de computadoras. A menudo son las primeras y fundamentales experiencias en el aprendizaje de PC porque requieren posiblemente la menor cantidad de demanda cognitiva y conocimiento técnico. El propósito de las experiencias desconectadas es introducir conceptos preliminares y superpuestos relacionados con el PC. Nishida et al. (2009) aconseja a los profesores que se centren en actividades dirigidas por los estudiantes que sean fáciles de implementar. Los maestros pueden usar juegos para enseñar o usar métodos de gamificación para integrar su enseñanza con desafíos y recompensas.
- **Las experiencias de retoque** involucran actividades que separan las cosas, cambian o modifican objetos existentes. Estos objetos pueden ser bloques de construcción, rompecabezas, simulaciones digitales o electrónicas, código de programación, etc. Los estudiantes no están construyendo un objeto, digital o de otro tipo, exploran los cambios en los objetos existentes y luego consideran las implicaciones de los cambios. El objetivo es proporcionar un contexto para explorar modificaciones incrementales, sin el desafío adicional de construir realmente el objeto.
- **Hacer** experiencias implican actividades donde la construcción de nuevos objetos es el enfoque principal. Al hacer experiencias, los estudiantes están obligados a: resolver problemas, hacer planes, seleccionar herramientas, reflexionar, comunicar, prototipar y probar, hacer conexiones entre conceptos.

- **El remezcla** se refiere a experiencias que implican la apropiación de objetos o componentes de aquellos con el objetivo de hacer uso de él para otros fines o en otros objetos. El remezcla implica modificar o adaptar objetos de alguna manera, y requiere un nivel significativo de competencia para identificar un objeto utilizable y luego adaptarlo y modificarlo.

En resumen, el marco desarrollado por Kotsopoulos et al. (2017) describe principalmente las actividades que un estudiante puede emprender para aprender el pensamiento informático.

La lista abarca desde las experiencias más simples — desenchufadas y retocadas hasta aquellas que requieren más perspectiva, mejores conocimientos de PC procesados — haciendo y remezclando.

Sin embargo, las actividades mencionadas no tienen que aplicarse solo al aprendizaje de ciencias de la computación. Como mostraremos en la siguiente sección, el PC puede tratarse como una habilidad como el pensamiento crítico o la resolución de problemas que se puede aprender aplicándola a varias materias escolares como parte de un enfoque transversal.

### 2.3 Pensamiento Computacional en Educación Obligatoria: ¿Cómo incluir PC en los currículos escolares?

*«La idea subyacente en el pensamiento computacional es desarrollar modelos y simulaciones de problemas que uno está tratando de estudiar y resolver».*

- *Dave Moursund*

En la sociedad del conocimiento, las organizaciones operan en el contexto de una economía global caracterizada por una intensa competencia, interdependencia económica y colaboración (Van Laar, et al., 2017). Los empleados necesitan habilidades suficientes para adaptarse a los requisitos cambiantes de sus trabajos. La pregunta mayoritariamente preguntada es: ¿Está el alumno habilitado para poner en práctica los conceptos? (Ahmad,

Karim, Din & Albakri, 2013; Carnevale & Smith, 2013). Esto significa que la adaptación tiene un gran significado en el siglo XXI. Los investigadores, al nombrar tales habilidades del siglo XXI, a menudo incluyen la colaboración, la comunicación, la alfabetización digital, la ciudadanía, la resolución de problemas, el pensamiento crítico, la creatividad y la productividad (Voogt & Roblin, 2012).

El pensamiento computacional es un término relativamente nuevo, pero mientras los investigadores comenzaron a prestarle más atención, quedó muy claro que el PC es la habilidad del siglo XXI (Black et al., 2009). El PC permite a los científicos no informáticos beneficiarse de un enfoque computacional para la resolución de problemas (Cuny et al., 2010). Les ayuda a entender los problemas que son computables, para determinar las herramientas y métodos correctos para resolver ciertos problemas, así como ayuda a la exploración de las limitaciones del método. Casi todas las disciplinas han sido influenciadas por el pensamiento computacional de alguna manera, tanto en las ciencias como en las humanidades (Mohaghegh, McCauley, 2016).

Tenemos que asumir que muchos de los estudiantes de hoy en día tienen contacto diario con las computadoras y el mundo virtual. Aprender cómo operar los dispositivos y cómo usarlos es una habilidad básica, PC es un enfoque más específico donde se enseña a los estudiantes a operar dentro y fuera del mundo digital. PC permite a los estudiantes pensar de una manera diferente, expresarse a través de varios medios, resolver problemas y analizar los problemas cotidianos de una manera organizada.

La alfabetización digital no debe confundirse con el Pensamiento Computacional, sin embargo, esos dos términos se superponen. La alfabetización digital es principalmente el estudio de cómo los estudiantes pueden utilizar de manera competente y segura las herramientas y recursos digitales. Por el contrario, el PC es una competencia que pueden ser adquiridas por aquellos que ya están digitalmente alfabetizados. Es un

concepto más complejo que va más allá del uso de computadoras o programación. Al desarrollar habilidades de PC, los estudiantes pueden desarrollar simultáneamente habilidades de pensamiento lógico y habilidades de resolución de problemas, así como aprender a codificar y programar. Ioannidou, Bennett, Repenning, Koh, & Basawapatna (2011) afirman que el aprendizaje de habilidades de PC incluye:

- formular problemas para el uso de computadoras con el fin de facilitar la solución;
- organizar y analizar lógicamente los datos;
- representar datos a través de abstracciones;
- automatización de soluciones a través de procesos algorítmicos;
- identificar, analizar e implementar posibles soluciones como la combinación más eficiente y eficaz de medidas y recursos; y
- generalizar y transferir este proceso a una variedad de áreas problemáticas. Hemos fusionado y enfatizado estos componentes (computacionales) en nuestra progresión de aprendizaje

Cuando hablamos de incluir habilidades de PC en el plan de estudios de la escuela, podemos tomar dos enfoques:

- colocación de PC dentro de un tema específico como Matemáticas o Ciencias de la Computación como una actividad independiente. El PC es un objetivo de aprendizaje en este enfoque.
- enfoque holístico que asume PC tiene el potencial de ser utilizado en una práctica interdisciplinaria en diferentes campos científicos. El enfoque holístico requiere que los maestros comparen y contrasten las aplicaciones de PC en diferentes situaciones. A continuación, los estudiantes son empujados a anticipar otras situaciones en las que se puede aplicar PC.

Como dijo Lockwood, Money, 2017:

*«Para algunas escuelas, tal vez no sea posible ejecutar un plan de estudios en el que se pueda enseñar PC por separado. Como tal, es importante que se hagan esfuerzos para garantizar que el PC no solo se enseña en lo que respecta a las ciencias de la computación, sino que también puede incorporarse a muchas otras áreas de la educación. Incluso si es posible enseñar PC directamente, es importante que los estudiantes sean conscientes y muestren cómo las habilidades y el conocimiento de PC se pueden aplicar a muchas áreas».*

Al involucrar a los estudiantes en el pensamiento computacional a través de múltiples dominios, mientras usan las mismas construcciones de programación y el mismo entorno de modelado, los maestros pueden permitir a los estudiantes darse cuenta de las similitudes entre dominios en términos de las construcciones y prácticas computacionales/matemáticas subyacentes. La inmersión significativa en experiencias de aprendizaje auténticas ocurre durante un período de varios años (Lehrer et al., 2008). Por lo tanto, el uso del pensamiento informático para resolver y comprender problemas que están presentes en dominios no relacionados con Ciencias de la Computación o Matemáticas solo ayudará a los estudiantes a comprender mejor este concepto y desarrollar habilidades más significativas.

Según Kolodner [p 57] *“...es importante no caer presa de la noción equivocada de que, si uno aprende habilidades de pensamiento computacional en un contexto, uno será automáticamente capaz de usarlas en otro contexto. Más bien, será importante recordar que uno puede aprender a usar habilidades de pensamiento computacional a través de contextos solo si pueden usar las mismas habilidades (y cómo lo harían)”.*

El PC puede profundizar la comprensión en diversos campos científicos como la química, la biología, las ciencias sociales, las lenguas extranjeras, etc. Como argumentó Resnick (2011), la mayoría de las personas trabajan mejor en cosas que les importan y que son significativas para ellos, por lo que integrar el

estudio de la abstracción en actividades concretas ayuda a hacerla significativa y comprensible. Stewart y Golubitsky (1992) señalan que, independientemente del dominio, el trabajo de los científicos implica la construcción y refinación de modelos del mundo. Por lo tanto, la construcción de modelos a través de la programación corresponde a la práctica científica básica. Según Sengupta, Kinnebrew, Biswas y Clark (2012) los estudios han demostrado que a través de la programación, los estudiantes pueden construir representaciones de conceptos físicos como la velocidad y la aceleración, y lograr una comprensión más profunda de estos conceptos.

#### Ejemplos de aplicación del PC en todos los temas:

Física	Las prácticas de PC son fundamentales para el desarrollo de la experiencia en una variedad de disciplinas STEM. Sengupta, Kinnebrew, Biswas y Clark (2012) diseñaron un entorno de aprendizaje que apoyó el aprendizaje de PC a través del modelado y la simulación. Su producto se centró en modelar fenómenos mecánicos newtonianos como la trayectoria, velocidad y aceleración de bolas colocadas en un entorno específico. Los estudiantes podrían influir en este entorno y observar cómo reacciona.
Geografía	Anuar, Mohamad y Minoi (2021) demostraron que a través del arte los estudiantes pueden captar conceptos esenciales del pensamiento computacional como la abstracción, la descomposición y los algoritmos. El estudio utilizó el aprendizaje basado en problemas para diseñar actividades de pensamiento computacional para 22 estudiantes de 10 a 11 años. La habilidad de abstracción se entrenó a través de dibujar y colorear el mapa itinerante con el etiquetado adecuado. La habilidad de descomposición se entrenó

	<p>cuando se pidió a los estudiantes que esbozaran un mapa de viaje simplificado, y se requería un pensamiento algorítmico cuando los estudiantes tenían que dibujar un punto para cada ubicación en la cuadrícula y escribir cada ruta como un algoritmo. Los participantes tuvieron que organizar y analizar lógicamente los datos, y luego representar estos datos a través de abstracciones (dibujos de un edificio, carretera, etc.).</p>
Literatura	<p>Cabo y Lansiquot (2016) crearon un curso: Programación de Narrativas: Narración animada por ordenador (<i>Narrativas de programación: Computer Animated Storytelling (PN)</i>), diseñado para ayudar a los estudiantes universitarios a desarrollar habilidades de pensamiento computacional a través de la programación informática y el desarrollo de habilidades de escritura. Dos disciplinas distintas de inglés y Ciencias de la Computación estaban conectadas para los propósitos de este curso. El objetivo del estudiante era crear un prototipo de videojuegos basado en la narrativa. Los estudiantes luego crearon historias como un programa de computadora. Se introdujeron en el uso de técnicas de diagrama de flujo y estructuras de programación, como secuenciación, bucles de repetición y declaraciones de decisión, para resolver un problema con un algoritmo, a conceptos de programación orientada a objetos, como clases, objetos, propiedades y métodos.</p>

Anteriormente, demostramos cómo el pensamiento computacional se puede aplicar a varios campos del aprendizaje. Sin embargo, al pensar en actividades que utilizan PC, no debe olvidarse que es igualmente importante adaptarlas al nivel de habilidad de los estudiantes, así como a su edad. Diseñar un juego o

crear una simulación son tareas que requieren un cierto nivel de habilidades y conocimientos. En los siguientes ejemplos de cómo se pueden introducir lecciones de pensamiento computacional integrado en la escuela.

Milks, et al., (2021) dan ejemplos de computación integrada en el currículo K-12 ordenado por edad. Sugieren las siguientes actividades:

**Para el grupo de edad de 4 a 5 años:**

- Divida la tarea de crear un objeto en pasos más pequeños. Un objeto se puede hacer de arcilla de modelado, papel, cuentas, etc.
- Realiza una secuencia de notas musicales o movimientos de baile.
- Use dados para seleccionar una acción o sonido para hacer (por ejemplo, sonido de rana si rueda un 3, sonido de pollo si rueda 5). Otros dados se pueden utilizar para determinar el número de veces para repetir esa acción o sonido.
- Cree gráficos de datos relevantes para sus estudiantes o aulas (por ejemplo, cumpleaños).
- Ordenar objetos según atributos compartidos (por ejemplo, reino animal, grupos de alimentos).
- Dibuja cómo se ve un árbol en diferentes épocas del año.
- Cree un mapa y proporcione instrucciones precisas para que un robot viaje a diferentes destinos en el mapa.

**Para el grupo de edad de 8 a 11 años:**

- Componer una canción usando una herramienta computacional como Scratch
- Identifique la gran tarea de hacer un árbol de masa de juego, identifique tareas más grandes y más pequeñas, como hacer un tronco, hojas y manzanas.

- Ponga la trama de una historia en el orden correcto.
- Divida una rutina diaria en partes más pequeñas y haga un libro pequeño para contar la historia de su propia rutina.
- Utilice Scratch para crear programas que calculan el área y el perímetro dados las entradas del usuario.
- Desarrollar historias de diferentes perspectivas históricas utilizando datos de fuentes primarias.
- Use una simulación para explorar fenómenos científicos (por ejemplo, cómo la masa afecta la fuerza cuando dos objetos chocan).

**Para el grupo de edad de 11 a 18 años:**

- Cree una visualización que haga hincapié en el sesgo o la injusticia con un conjunto de datos.
- Desarrollar un algoritmo para descomponer un mensaje de ensayo
- Utilizar para decodificar o remezclar un modelo computacional sobre un fenómeno científico (por ejemplo, el clima).
- Analizar y evaluar el sesgo en datos de segunda mano sobre cuestiones sociocientíficas (por ejemplo, cáncer, contaminación).
- Analizar los datos examinando críticamente las estructuras sociopolíticas dentro de la sociedad y la tecnología (por ejemplo, redlining y digital redlining).

En resumen, la informática impregna todas las disciplinas y esferas de conocimiento, por lo que se podría argumentar que el pensamiento computacional, incluida la aplicación de conceptos y habilidades informáticas para resolver problemas, no solo en Ciencias de la Computación sino también en otras disciplinas, debería formar parte de la educación del siglo XXI utilizada como herramienta para una amplia gama de estudiantes universitarios, incluidos los que no se especializan en informática (Guzdial, 2008; Wing, 2006).

## 2.4 Modelo de Pensamiento Computacional para la Educación Online

### 2.4.1 Introducción

El aprendizaje en línea fue invaluable durante la pandemia. Los maestros se han familiarizado con las herramientas de enseñanza en línea, y asumimos que usarán algunos de ellos de una forma u otra en el futuro. Es por eso que, independientemente de la situación epidémica, el aprendizaje en línea está aquí para quedarse.

La forma de aprendizaje que un maestro elige puede basarse en el contexto escolar o en las preferencias personales. El modo de enseñanza seleccionado puede determinar las actividades realizadas en clase y cómo se supervisa el progreso y se evalúan los resultados.

- **El aprendizaje en línea** se lleva a cabo con la ayuda de Internet en un entorno totalmente en línea. Estudiantes y profesores de diferentes ubicaciones físicas interactúan entre sí a través de una conexión a Internet. Los seminarios web, las sesiones de aula virtuales y las colaboraciones en línea de pizarra son formas de aprendizaje en línea.
- **El aprendizaje en el sitio** requiere que los maestros y estudiantes estén físicamente presentes para la lección. Significa que los maestros se comunican y comparten materiales de aprendizaje con los estudiantes en un aula física. Este modo de aprendizaje es limitado geográficamente y menos accesible que el aprendizaje en línea.
- **El aprendizaje híbrido** es un enfoque de la educación que combina actividades mediadas por computadora y materiales educativos en línea con métodos tradicionales de clase basados en el lugar. En este contexto, los estudiantes aprenden a través de una mezcla de actividades en persona y en línea. El aprendizaje híbrido apoya y refuerza el aprendizaje in situ. Los estudiantes pueden acceder a materiales como conferencias, lecturas y foros de discusión en cualquier momento y aprender a su propio ritmo. También es un método implementado en una situación en la que algunos de los

estudiantes no pueden estar presentes en el aula física por alguna razón (por ejemplo, problemas de salud o problemas de viaje).

El aprendizaje en línea es similar al aprendizaje in situ, pero a menudo requiere el conocimiento de diferentes herramientas y métodos de participación. Hemos incluido descripciones de diversos modos de enseñanza en el contexto de nuestro modelo de Pensamiento Computacional para la Educación Online para mostrar posibilidades y diferencias entre los diferentes entornos de aprendizaje. En los siguientes subcapítulos, describiremos cómo facilitar el aprendizaje en un entorno en línea o híbrido, cómo seleccionar herramientas para el aprendizaje en línea, cómo planificar e implementar actividades para desarrollar habilidades de PC.

#### 2.4.2 Funciones y responsabilidades de los docentes en la facilitación del aprendizaje en línea

Bosch (2016) señala seis objetivos pedagógicos básicos del aprendizaje: contenidos, apoyo social y emocional, cuestionamiento, fomento de la reflexión, fomento del aprendizaje colaborativo y evaluación.

Hay muchas maneras en que el contenido se puede entregar y presentar en un entorno en línea o un entorno híbrido. Los profesores pueden utilizar sistemas de gestión del aprendizaje (CMS/LMS) tales como: Blackboard, Canvas o Moodle. Una vez que se decide qué herramientas usar para enseñar la lección y compartir los materiales de aprendizaje con los estudiantes, hay una necesidad de centrarse en cómo se debe enseñar la lección.

Sousa (2016, p.17) describe el compromiso de los estudiantes como *«la cantidad de atención, interés, curiosidad y conexiones emocionales positivas que los estudiantes tienen cuando están aprendiendo, ya sea en el aula o por su cuenta»* (2016, p. 17).

Como señala Bosch, es importante apoyar a los estudiantes social y emocionalmente. Los estudiantes necesitan a alguien con quien hablar, ya

sea para ayudarlos a comprender un concepto complejo o para proporcionar consejos que, naturalmente, nos llevan a discutir qué es una comunidad de aprendizaje y por qué construirlo es importante en un entorno en línea.

El término «comunidad de aprendizaje» puede significar una variedad de cosas, pero en el nivel más básico, la comunidad es gente que interactúa entre sí. La comunidad se puede construir tanto en la configuración presencial como en línea. Con la aparición de la COVID-19, los educadores tuvieron que involucrar a los estudiantes en entornos de aprendizaje en línea, y esta situación separó a los estudiantes de sus profesores y compañeros, creando una amenaza para el bienestar emocional y el desarrollo social. Es por eso que es importante enfatizar que la construcción de la comunidad es posible y necesaria en un entorno en línea, y es parte del papel de un maestro para facilitar las interacciones entre los estudiantes.

Estimular la discusión entre grupos haciendo las preguntas «correctas» puede ayudar a los estudiantes a pensar críticamente sobre un tema o tema. El método socrático puede ser particularmente útil para involucrar y desafiar a los estudiantes intelectualmente para que entiendan los problemas difíciles. Las actividades pedagógicas que requieren que los estudiantes reflexionen sobre lo que aprenden y compartan sus reflexiones con sus profesores y compañeros amplían y enriquecen la experiencia de aprendizaje.

También se anima a involucrar a los estudiantes en proyectos grupales, ya que el aprendizaje colaborativo es una herramienta esencial para crear conocimiento, así como para fomentar las relaciones entre estudiantes, generando revisión y evaluación por pares (Fredericksen, 2015).

El marco de las Comunidades Académicas de Compromiso (Academic Communities of Engagement (ACE)) (Borup et al., 2014); Borup, Graham, et al., 2020; Borup, Jensen, et al., 2020) se tiene en cuenta la presencia de tres roles comunitarios diferentes. Maestros, compañeros y padres en un proceso de aprendizaje como parte de la comunidad de aprendizaje. Los autores del

marco afirman que cuanto más comprometidos y presentes son los actores de la comunidad, más estudiantes participan en el proceso de aprendizaje.

ACE Framework también identifica las responsabilidades de los profesores que incluyen:

- ❖ **Orientación:** ayudar a los estudiantes a comprender las expectativas, los sistemas y las estrategias para aprender en línea
- ❖ **Instruyendo:** proporcionar a los estudiantes comentarios y tutorías que afectan directamente su comprensión del plan de estudios del curso.
- ❖ **Organización y diseño:** proporcionar a los estudiantes un entorno de aprendizaje y actividades de aprendizaje que fomenten el aprendizaje
- ❖ **Nutrir:** establecer relaciones estrechas y cariñosas entre el estudiante y el maestro
- ❖ **Facilitar la comunicación:** fomentar la comunicación con y entre los estudiantes, los padres y otras partes interesadas
- ❖ **Seguimiento y motivación:** seguir el progreso de los estudiantes y motivarlos a participar plenamente en actividades de aprendizaje (Borup, Graham, & Drysdale, 2014; Borup & Stevens, 2016, 2017).

En cuanto al proceso de evaluación en sí, hay muchas herramientas disponibles en línea para crear pruebas, ensayos y encuestas. Sin embargo, también hay muchos artefactos que los estudiantes pueden crear y evaluar, los ejemplos incluyen: presentaciones en el aula, videos y podcasts de YouTube, discusiones semanales en las clases en foros de discusión o publicaciones de blog que se pueden revisar una y otra vez para examinar cómo los estudiantes han participado y progresado con el tiempo. También son más útiles para los instructores para evaluar su propia enseñanza y para revisar lo que funcionó y lo que no funcionó en una clase. A diferencia del trabajo en grupo presencial que normalmente terminaba en el escritorio del instructor, el contenido generado en línea se puede compartir con otros más allá del aula (Bosh, 2016).

Para resumir, con el fin de desarrollar los conocimientos y habilidades de los estudiantes, un profesor necesita elegir las estrategias de enseñanza óptimas. Es importante crear objetivos de aprendizaje claros e informar a los estudiantes sobre ellos. Además, mientras se trabaja en un entorno online, es fundamental conocer las posibilidades de las herramientas que un profesor planea utilizar en el aula virtual. Adquirir conocimiento es un proceso con muchos obstáculos, sin embargo, con una buena organización del trabajo y un ambiente saludable y positivo para el aprendizaje, los estudiantes pueden aprender más eficazmente y comprender mejor los contenidos discutidos. Al construir buenas relaciones estudiante-maestro, estudiante-peer e involucrar a los padres en el proceso de aprendizaje (especialmente con la educación híbrida y en línea), es probable que veamos resultados positivos y duraderos.

Supervisar el progreso y proporcionar retroalimentación también es esencial para diseñar un entorno de aprendizaje eficaz. Saber qué tan bien entienden los estudiantes los conceptos discutidos, y cuán efectivas son sus habilidades de implementación, permite al profesor evaluar la estrategia educativa. A su vez, un estudiante que es capaz de monitorear su progreso está motivado a continuar la mejora de sus habilidades y conocimientos.

#### **2.4.3 Planificación e implementación de actividades para desarrollar habilidades de PC**

Conociendo las responsabilidades y funciones de un profesor en un entorno en línea, ahora podemos pasar a planificar actividades destinadas a desarrollar habilidades de pensamiento informático.

Antes de comenzar a discutir actividades específicas, discutiremos brevemente el proceso de planificación de actividades, específicamente el proceso de diseño del plan de lecciones. El plan de lecciones puede ser una guía útil para un maestro, debe reflejar la filosofía de enseñanza y, lo que es más importante, los objetivos de aprendizaje del maestro (Jensen 2001; Nesari y Heidari 2014).

Un plan de lecciones generalmente implica información como: *grado, asignatura, duración de la lección, fecha, tema de la lección, objetivos de aprendizaje, actividades de enseñanza y aprendizaje, asignación de tiempo de las actividades, recursos didácticos y actividades de evaluación*. También refleja las creencias de un maestro, la comprensión y la orientación hacia el currículo, la asignatura que se está enseñando y la pedagogía (Li y Zou 2017; Shen et al. 2007). Al comienzo de este capítulo, presentamos el marco de Brennan y Resnick. Este marco sirve de base para el plan de estudios o plan de acción a largo plazo. Un plan de lecciones, sin embargo, tiene como objetivo establecer metas mensurables a corto plazo que nos permitan tener control sobre el proceso de aprendizaje de los estudiantes y la calidad y ajuste de los métodos educativos utilizados. Los objetivos o metas de aprendizaje son cruciales para el proceso de diseño de actividades educativas. Los objetivos claros ayudan a los estudiantes a entender el propósito del aprendizaje. Ayudan al maestro a planificar e impartir instrucción a un nivel apropiado, y también ayudan con las evaluaciones y la evaluación de las actividades de aprendizaje más adelante.

Planificación de la lección de antemano:

- aumenta la productividad
- permite evitar la enseñanza fuera de contexto/desperdicio de tiempo
- permite a los profesores organizar la información que se presentará a los alumnos
- permite establecer metas claras sobre lo que el profesor quiere lograr
- los estudiantes saben en qué van a ser evaluados
- permite al profesor evaluar si se cumplieron los objetivos de aprendizaje
- permite colocar la lección en el interior política curricular

Para crear un plan de lecciones, considere:

1. ¿Cómo encajan las actividades de PC en el plan de estudios?

2. ¿Cómo evaluar qué conocimientos y habilidades necesita un estudiante para practicar PC?
3. ¿Cuánto tiempo durará cada actividad?
4. ¿Qué herramientas de enseñanza usarás, cómo (puedes usar programas como Scratch, Alice, AgentSheets, etc.)?
5. ¿Cómo diseñar actividades que se correspondan con las habilidades y conocimientos actuales de un estudiante?
6. ¿Cómo hacer la evaluación y la evaluación?

Con el fin de planificar eficazmente los objetivos de aprendizaje para el pensamiento computacional, necesitamos planificar conscientemente actividades fáciles al principio y pasar lentamente a las cada vez más difíciles. Para entender este proceso y planificar los objetivos de aprendizaje, podemos confiar en la taxonomía de Bloom, que divide los objetivos de aprendizaje en niveles de complejidad y especificidad.

Bloom menciona seis objetivos educativos que describen los procesos cognitivos que los estudiantes deben usar para aprender, estos son: Recuerda, entiende, aplica, analiza, evalúa, crea. Los estudiantes deben abordar un tema o tema desde el nivel más bajo (recuerde) antes de pasar a niveles más altos de pensamiento.

Recordar implica reconocer y recordar, así como describir, identificar, enumerar. La comprensión requiere acciones tales como interpretar, parafrasear, explicar, clasificar, resumir o comparar. La aplicación ocurre cuando se les pide a los estudiantes que usen, resuelvan, predigan, apliquen o demuestren una tarea o acción, es probable que trabajen en este nivel de pensamiento. El análisis ocurre cuando los estudiantes pueden establecer conexiones entre ideas y utilizar sus habilidades de pensamiento crítico. Los profesores pueden pedir a los estudiantes que diferencien, comparen, deconstruyan hechos o datos. La evaluación ocurre cuando los estudiantes prueban, verifican o critican artefactos. Finalmente, crear es el nivel más alto

de aprendizaje en la taxonomía de Bloom, requiere planear y producir algo. Los estudiantes pueden tener que escribir, diseñar, combinar, idear o modificar algo.

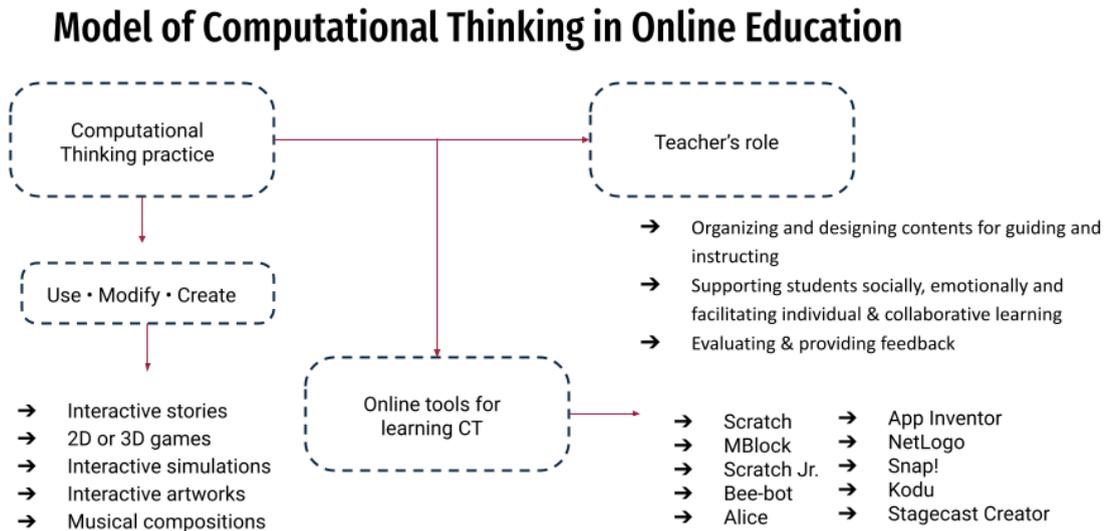
Como se postula en la taxonomía de Bloom, para crear actividades de aprendizaje efectivas es necesario profundizar la comprensión del estudiante, luego, lenta y gradualmente, involucrar a los estudiantes en actividades cada vez más complejas. Podemos ver la misma filosofía en el Marco USE-MODIFY-CREATE (Lee et al., 2011), que tiene como objetivo demostrar cómo se debe enseñar el pensamiento computacional. El marco USO-MODIFICAR-CREAR como su nombre indica, consta de tres fases, que describiremos brevemente a continuación:

- **USO:** Durante esta fase, los estudiantes aprenden a usar los programas y proyectos existentes que otros han realizado. Esto puede implicar realizar operaciones de tutoriales con guiones y explorar nuevo software.
- **MODIFICAR:** A medida que se gana comodidad en el uso de las herramientas, los estudiantes pueden comenzar a experimentar modificando los programas o proyectos existentes, haciendo contribuciones cada vez más originales. Durante esta fase, los estudiantes comienzan a entender cómo pueden controlar los mecanismos subyacentes para lograr diferentes resultados.
- **CREAR:** En esta fase, los estudiantes aplican sus habilidades de crecimiento para crear un producto original. Este trabajo mostrará niveles crecientes de abstracción y automatización que pueden haber estado presentes en actividades anteriores.

En resumen, en la teoría antes mencionada pasamos de la actividad más simple (usando una herramienta preparada) a la actividad más difícil: creación de artefactos para evaluación y evaluación. Por ejemplo, el profesor

puede pedir a los estudiantes que creen una simulación, una historia interactiva, un juego o una animación.

## 2.5 Resumen



Como Lee et al. (2021) declarado anteriormente, el pensamiento computacional se puede enseñar a través de tres actividades: usar, modificar, crear.

Los estudiantes pueden crear varios artefactos que luego serán evaluados. Estos artefactos pueden tomar la forma de *historias interactivas, juegos, simulaciones, obras de arte o composiciones musicales*. Los estudiantes también pueden usar y modificar los códigos que otros han hecho.

Un número significativo de programas para el aprendizaje de PC están disponibles en línea. Entre los más populares se encuentran Scratch, Scratch Jr, Alice, AgentCubes, App Inventor. Discutimos las funciones de estos programas en el tercer capítulo de este Manual.

Además de los programas destinados a desarrollar el pensamiento computacional, el profesor también tiene a su disposición una serie de programas que le permitirán proporcionar instrucción en línea, organizar

actividades grupales en línea y recopilar comentarios de las actividades realizadas en forma digital. También hay herramientas para la comunicación entre los participantes, por ejemplo, foros, redes sociales y salas de chat que permiten a los estudiantes tener una discusión y compartir reflexiones. Son vehículos para discusiones grupales y para compartir reflexiones. Por último, existen herramientas para cuestionarios y encuestas que son útiles para la retroalimentación y la evaluación.

En nuestro modelo, tratamos de combinar el modelo de enseñanza en línea con el modelo de enseñanza del pensamiento informático. Conscientes de que el fomento del pensamiento computacional es posible principalmente a través del uso de herramientas disponibles en forma digital, describimos los detalles del trabajo de un profesor en un entorno online y los detalles, actividades y productos generados en el aprendizaje del pensamiento computacional.

## Verificación de conocimientos

Responda a las siguientes preguntas a su mejor capacidad:

- ¿Cómo podemos definir el PC como una habilidad clave del siglo XXI para los niños en edad escolar?

.....  
.....

- ¿Cómo se puede aplicar el PC en un tema que no es Matemáticas o Ciencias de la Computación?

.....  
.....

- ¿Qué se necesita para seguir desarrollando la agenda del PC en los entornos de educación obligatoria?

- 
- 
- ¿Qué significa evaluar el PC?
- 
- 

## **Referencias:**

Ahmad, M., Karim, A. A., Din, R., & Albakri, I. S. M. A. (2013). Assessing ICT competencies among postgraduate students based on the 21st century ICT competency model. *Asian Social Science*, 9(16), 32e39.

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K- 6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge. *Journal of Educational Technology and Society*, 19(3), 47-57. <http://A K-6 Computational Thinking Curriculum Framework>

Anuar, N. H., Mohamad, F. S., & Minoi, J.-L. (2021). Art-Integration in Computational Thinking as an Unplugged Pedagogical Approach at A Rural Sarawak Primary School. *International Journal of Academic Research in Business and Social Sciences*, 11(14), 21–39.

Allan, W., Coulter, B., Denner, J., Erickson, J., Lee, I., Malyn-Smith, J., & Martin, F. (2010). Computational thinking for youth. White Paper for the ITEST Small Working Group on Computational Thinking (CT).

Borup, J., Graham, C. R., & Drysdale, J. S. (2014). The nature of teacher engagement at an online high school. *British Journal of Educational Technology*, 45(5), 793–806. <https://doi.org/10.1111/bjet.12089>

Borup, J., West, R. E., Graham, C. R., & Davies, R. S. (2014). The Adolescent Community of Engagement: A framework for research on adolescent online learning. *Journal of Technology and Teacher Education*, 22(1), 107–129.

Borup, J., & Stevens, M. A. (2016). Parents' perceptions of teacher support at a cyber charter high school. *Journal of Online Learning Research*, 2(3), 227–246.

Borup, J., & Stevens, M. A. (2017). Using student voice to examine teacher practices at a cyber charter high school. *British Journal of Educational Technology*, 48, 1119-1130. <https://doi.org/10.1111/bjet.12541>

Bosch, C. (2016). Promoting Self-Directed Learning through the Implementation of Cooperative Learning in a Higher Education Blended Learning Environment. Johannesburg, SA: Doctoral dissertation at North-West University.

Branch, R. M., & Kopcha, T. J. (2014). Instructional design models. In Handbook of research on educational communications and technology (pp. 77-87). Springer, New York, NY.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Paper presented at the American Educational Research Association. Canada: British Columbia.

Carnevale, A. P., & Smith, N. (2013). Workplace basics: The skills employees need and employers want. Human Resource Development International, 16(5), 491e501. <http://dx.doi.org/10.1080/13678868.2013.821267>

Cabo, C., & Lansiquot, R. (2016). Integrating creative writing and computational thinking to develop interdisciplinary connections.

Cuny, J., Snyder, L., & Wing, J. M. (2010). Demystifying computational thinking for non-computer scientists. Unpublished manuscript in progress, referenced in <http://www.cs.cmu.edu/link/research-notebookcomputational-thinking-what-and-why>

Curzon, P., Black, J., Meagher, L. R., & McOwan, P. (2009). cs4fn.org: Enthusing students about Computer Science. Proceedings of Informatics Education Europe IV, 73-80.

Fredericksen, E. (February 4, 2015). Is online education good or bad? And is this really the right question? The Conversation. Retrieved from: <https://theconversation.com/is-onlineeducation-good-or-bad-and-is-this-really-the-right-question-35949>

Guzdial, M. (2008). Paving the way for computational thinking. Communications of the ACM, 51(8), 25-27.

How Online Education Aids Distance Learning. (2022, January 27). Infinity Learn.

<https://infinitylearn.com/surge/blog/general/how-online-education-aids-distance-learning/>

How Computational Thinking Sets Kids Up for Success In and Outside of the Classroom. (2021, June 14). CodeWizardsHQ.

<https://www.codewizardshq.com/computational-thinking/>

Ioannidou, A., Bennett, V., Repenning, A., Koh, K. H., & Basawapatna, A. (2011). Computational Thinking Patterns. Online Submission.

ISTE. (2016). ISTE (International Society for Technology in Education) Standards for Students. Eugene, OR.

ISTE, CSTA. (2011). Computational Thinking in K–12 Education Leadership Toolkit.

Jensen, L. 2001. "Planning Lessons." In Teaching English as a Second or Foreign Language, edited by M.M. Celce. 3rd ed, pp. 403–409. Boston: MA: Heinle & Heinle.

Koc, S., & Goodell, J. E. (2018). Preparing teachers for mobile learning applications grounded in research and pedagogical frameworks. In Handbook of Research on Mobile Devices and Smart Gadgets in K-12 Education (pp. 103-114). IGI Global.

Kola, M. (2021). Pre-service teachers' action research: technology education lesson planning in a South African University. Educational Action Research, 29(1), 99-117.

Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A pedagogical framework for computational thinking. Digital Experiences in Mathematics Education, 3(2), 154-171.

Lehrer, R., Schauble, L., and Lucas, D., (2008). Supporting development of the epistemology of inquiry. Cognitive Development, 23 (4), 512-529.

Li, W., and W. Zou. 2017. "A Study of EFT teacher expertise in lesson planning." Teaching and Teacher Education 66: 231–241.

Mills, K., Coenraad, M., Ruiz, P., Burke, Q., & Weisgrau, J. (2021). Computational Thinking for an Inclusive World: A Resource for Educators to Learn and Lead. Digital Promise.

Mohaghegh, D. M., & McCauley, M. (2016). Computational thinking: The skill set of the 21st century.

Nesari, A.J., and M. Heidari. 2014. "The Important Role of Lesson Plan on Educational Achievement of Iranian EFL Teacher Attitudes." International Journal of Foreign Language Teaching & Research 3 (5): 25–31.

NRC 2011. Report of a Workshop on the Pedagogical Aspects of

Computational Thinking. 2011, National Research Council.

Sengupta, P., Kinnebrew, J., Biswas, G., Clark, D. (2012). Integrating Computational Thinking With K-12 Science Education - A Theoretical Framework. 4th International Conference On Computer Supported Education (CSEDU), 40-49.

Stewart, I. and Golubitsky, M. (1992). *Fearful Symmetry: Is God a Geometer?*, Blackwell Publishers, Oxford.

Van Laar, E., Van Deursen, A. J., Van Dijk, J. A., & De Haan, J. (2017). The relation between 21st-century skills and digital skills: A systematic literature review. *Computers in human behaviour*, 72, 577-588.

Voogt, J., & Roblin, N. P. (2012). A comparative analysis of international frameworks for 21st century competencies: Implications for national curriculum policies. *Journal of Curriculum Studies*, 44(3), 299-321. <http://dx.doi.org/10.1080/00220272.2012.668938>.

Smith, J. (2021, June 14) How Computational Thinking Sets Kids Up for Success In and Outside of the Classroom. From: CodeWizardsHQ. <https://www.codewizardshq.com/computational-thinking/>  
Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.

## Capítulo 3: Métodos de enseñanza en el pensamiento computacional

---

### 3.1 Enfoques pedagógicos actuales y estrategias para enseñar pensamiento computacional

El pensamiento computacional es uno de los conjuntos de habilidades más importantes que los educadores deben enseñar a sus jóvenes estudiantes. Es una piedra angular del desarrollo de la primera infancia, dando a los niños un enfoque para la resolución de problemas que desarrolla una solución y se

considera una de las competencias críticas para adaptarse al futuro (Hsu et al., 2018). En la última década, PC ha obtenido popularidad, tanto en el campo académico como en la práctica. El PC ha sido estudiada en diferentes países de todo el mundo. En consecuencia, las materias, los temas de investigación y las herramientas de enseñanza en el campo del PC se han desarrollado aún más (Hsu et al., 2018).

Sin embargo, a pesar de la creciente atención académica al tema, la realidad en muchas escuelas es que los maestros no están especializados en aplicar el PC en sus aulas. Esto se debe principalmente a la falta de formación docente en PC y la falta de conocimiento sobre cómo implementar la metodología porque todavía hay evidencia limitada en torno a los diversos desafíos que alguien debe tener en cuenta para diseñar experiencias de aprendizaje adecuadas y un plan de estudios basado en PC (Angeli & Giannakos 2020). Por lo tanto, existe una creciente demanda de material de apoyo que ayude a los maestros a enseñar a sus estudiantes diferentes materias basadas en el marco de PC. Esta sección tiene como objetivo presentar los enfoques pedagógicos actuales y las estrategias para enseñar PC. El PC todavía se utiliza principalmente en las asignaturas STEAM, pero el objetivo del proyecto es extender el PC a otras asignaturas y así hacer que el PC sea accesible a más profesores y estudiantes y especialmente debido a su adaptación a la educación en línea.

El objetivo es dar una visión general de las estrategias y metodologías actualmente disponibles que permiten el PC en la educación. Dado que nuestro enfoque se centra en ofrecer un marco para la implementación del PC en la educación en línea, los capítulos 3.1.1 y 3.1.2 presentarán enfoques para combinar estos cursos de PC y en línea. Por lo tanto, la estructura de esta sección es la siguiente: En primer lugar, se explicará el Modelo de Proceso de Diseño de Ingeniería en educación PC. Más adelante, se describirá la Metodología Flipped Learning o aprendizaje invertido.

### 3.1.1 Modelo de Proceso de Diseño de Ingeniería en Educación PC

El Marco Currículum de Massachusetts para la Ciencia y la Tecnología (2016) es un documento que aborda las directrices educativas obligatorias para los temas de Ciencia y Tecnología/ingeniería (STE) en este estado. Incluye la visión de la Educación en Ciencia y Tecnología/Ingeniería de sus autores — un panel de revisión de maestros y responsables políticos y el Grupo Asesor de Normas de Ciencia de la Próxima Generación (the Next Generation Science Standards (NGSS) Advisory Group) — así como una visión general de las prácticas de cada grado y los apéndices relacionados con el currículo/estándares y destaca la importancia de las matemáticas para medir, predecir y resolver problemas.

Este marco proviene de la necesidad de educar a los ciudadanos capaces de participar activamente en un mundo tecnológico, ya que el objetivo de la educación STE es *desarrollar ciudadanos científica y tecnológicamente alfabetizados que puedan resolver problemas complejos y multidisciplinarios y aplicar el razonamiento analítico y el pensamiento innovador a las aplicaciones del mundo real necesarias para la participación cívica, la preparación universitaria y la preparación profesional.*

La enseñanza y el aprendizaje están en el corazón de la educación científica y tecnológica/ingeniería de calidad. La visión de los estándares STE de Massachusetts es involucrar a los estudiantes en las ideas centrales a través de la integración de las prácticas de ciencia e ingeniería, mientras hacen conexiones con lo que saben y el mundo en el que viven. El objetivo de los Principios Rectores es ayudar a los educadores a crear programas STE relevantes, rigurosos y coherentes que apoyen el compromiso de los estudiantes, la curiosidad, el pensamiento analítico y la emoción por el aprendizaje a lo largo del tiempo. Los educadores, administradores y diseñadores de currículos pueden referirse a los Principios Rectores para desarrollar programas eficaces pre-K-12 STE.

<b>Énfasis en los Estándares STE</b>	<b>Implicación para el plan de estudios y la instrucción</b>
<b>Relevancia:</b> Organizado en torno a ideas explicativas básicas que explican el mundo que nos rodea	El objetivo de la enseñanza se centra en los estudiantes que analizan y explican los fenómenos y la experiencia
<b>El rigor:</b> Papel central de las prácticas de ciencia e ingeniería con conceptos	El aprendizaje basado en la investigación y el diseño implica un compromiso regular con prácticas para construir, usar y aplicar el conocimiento.
<b>Coherencia:</b> Ideas y prácticas se construyen a lo largo del tiempo y entre disciplinas	La enseñanza implica construir una historia coherente a lo largo del tiempo y entre disciplinas

Figura 1. DOE (2016). Cualidades de Ciencia y Tecnología/Educación de Ingeniería para Todos los Estudiantes.

Los principios rectores del Marco Curriculum de Massachusetts para la Ciencia y la Tecnología son:

### **Pertinencia**

1. Un programa eficaz de ciencia y tecnología/ingeniería desarrolla la capacidad de los estudiantes para aplicar sus conocimientos y habilidades para analizar y explicar el mundo que los rodea.
2. Un programa eficaz de ciencia y tecnología/ingeniería aborda los conocimientos previos y las ideas preconcebidas de los estudiantes.

Estas ideas se centran en la importancia que tienen los estudiantes de conocer el entorno para conexiones interesantes y trabajar en conceptos erróneos anteriores que van al núcleo de los conceptos.

### **Rigor**

3. La investigación, la experimentación, el diseño y la resolución de problemas analíticos son fundamentales para un programa eficaz de ciencia y tecnología/ingeniería.

4. Un programa eficaz de ciencia y tecnología/ingeniería ofrece oportunidades para que los estudiantes colaboren en los esfuerzos científicos y tecnológicos y comuniquen sus ideas.

5. Un programa eficaz de ciencia y tecnología/ingeniería transmite altas expectativas académicas para todos los estudiantes.

Esta visión trata de involucrar a los estudiantes en todo el proceso de compromiso y éxito mediante la colaboración activa con profesionales y trabajando en actividades y conceptos que serán importantes después de la vida escolar.

### **Coherencia**

6. Un programa eficaz de ciencia y tecnología/ingeniería integra el aprendizaje STE con las matemáticas y la alfabetización disciplinaria.

7. Un programa eficaz de ciencia y tecnología/ingeniería utiliza una evaluación regular para informar el aprendizaje de los estudiantes, guiar la instrucción y evaluar el progreso de los estudiantes.

8. Un programa eficaz de ciencia y tecnología/ingeniería involucra a todos los estudiantes, pre-K hasta el grado 12.

9. Un programa eficaz de ciencia y tecnología/ingeniería requiere una planificación coherente en todo el distrito y un apoyo continuo para su implementación.

Los autores manifiestan la idea de que, si se respetan todos estos puntos, el compromiso de los estudiantes aumentará y avanzará hacia una mejor ciudadanía.

El marco se basa en estándares de aprendizaje, que son resultados u objetivos, que reflejan lo que un estudiante debe saber y ser capaz de hacer. Cada

asignatura en cada grado tiene estándares que todos los estudiantes tienen que cumplir, por ejemplo:

**Ejemplo: Grado 1: Ciencias de la Tierra y del Espacio**

Asignatura/tema: Lugar de la Tierra en el Universo

1. Estándar: Use observaciones del Sol, la Luna y las estrellas para describir que cada uno parece elevarse en una parte del cielo, parece moverse a través del cielo y parece establecerse.
2. Estándar: Analizar los datos proporcionados para identificar las relaciones entre los patrones de cambio estacionales, incluidos los cambios relativos a la hora de amanecer y atardecer, la temperatura estacional y los patrones de lluvia o nevadas, y los cambios estacionales en el medio ambiente.

Declaración de aclaración:

Ejemplos de cambios estacionales en el medio ambiente pueden incluir cambios en el follaje, migración de aves y diferencias en la cantidad de actividad de los insectos.

### 3.1.2. El Modelo de Aula Flipped Basado en 5E en la educación de PC

Durante los últimos diez años, la metodología volteada ha crecido hasta convertirse en una de las principales tendencias en educación innovadora. El modelo de aprendizaje invertido/flipped learning promueve el aprendizaje centrado en el estudiante, que es el proceso de usar diferentes métodos de instrucción para dirigirse a las necesidades individuales, las motivaciones, el compromiso y los apoyos que permitirán a un estudiante tener éxito por su cuenta (Carnevale, 2017). Esta nueva forma de enseñanza ha sido definida por sus principales embajadores,

Jonathan Bergmann y Aaron Sams (2014), como:

*Básicamente, el concepto de una clase volteada es este: lo que se hace tradicionalmente en clase ahora se hace en casa, y lo que tradicionalmente se hace como tarea se completa ahora en clase.*

Sin embargo, la idea de las lecciones volteadas va mucho más allá y más profundo, ya que han sido capaces de probar más adelante:

*Flipped Learning es un enfoque pedagógico en el que la instrucción directa se mueve del espacio de aprendizaje grupal al espacio de aprendizaje individual, y el espacio grupal resultante se transforma en un entorno de aprendizaje dinámico e interactivo donde el educador guía a los estudiantes a medida que aplican conceptos y se involucran creativamente en la materia. (Red de Aprendizaje Flipped, 2014)*

La metodología Flipped se basa en los cuatro pilares:

## **F** Flexible Environment

Flipped Learning allows for a variety of learning modes; educators often physically rearrange their learning spaces to accommodate a lesson or unit, to support either group work or independent study. They create flexible spaces in which students choose when and where they learn. Furthermore, educators who flip their classes are flexible in their expectations of student timelines for learning and in their assessments of student learning.

<b>F.1</b>	<input type="checkbox"/> I establish spaces and time frames that permit students to interact and reflect on their learning as needed.
<b>F.2</b>	<input type="checkbox"/> I continually observe and monitor students to make adjustments as appropriate.
<b>F.3</b>	<input type="checkbox"/> I provide students with different ways to learn content and demonstrate mastery.

## L Learning Culture

In the traditional teacher-centered model, the teacher is the primary source of information. By contrast, the Flipped Learning model deliberately shifts instruction to a learner-centered approach, where in-class time is dedicated to exploring topics in greater depth and creating rich learning opportunities. As a result, students are actively involved in knowledge construction as they participate in and evaluate their learning in a manner that is personally meaningful.

L.1	<input type="checkbox"/> I give students opportunities to engage in meaningful activities without the teacher being central.
L.2	<input type="checkbox"/> I scaffold these activities and make them accessible to all students through differentiation and feedback.

## I Intentional Content

Flipped Learning Educators continually think about how they can use the Flipped Learning model to help students develop conceptual understanding, as well as procedural fluency. They determine what they need to teach and what materials students should explore on their own. Educators use Intentional Content to maximize classroom time in order to adopt methods of student-centered, active learning strategies, depending on grade level and subject matter.

I.1	<input type="checkbox"/> I prioritize concepts used in direct instruction for learners to access on their own.
I.2	<input type="checkbox"/> I create and/or curate relevant content (typically videos) for my students.
I.3	<input type="checkbox"/> I differentiate to make content accessible and relevant to all students.

## P Professional Educator

The role of a Professional Educator is even more important, and often more demanding, in a Flipped Classroom than in a traditional one. During class time, they continually observe their students, providing them with feedback relevant in the moment, and assessing their work. Professional Educators are reflective in their practice, connect with each other to improve their instruction, accept constructive criticism, and tolerate controlled chaos in their classrooms. While Professional Educators take on less visibly prominent roles in a flipped classroom, they remain the essential ingredient that enables Flipped Learning to occur.

P.1	<input type="checkbox"/> I make myself available to all students for individual, small group, and class feedback in real time as needed.
P.2	<input type="checkbox"/> I conduct ongoing formative assessments during class time through observation and by recording data to inform future instruction.
P.3	<input type="checkbox"/> I collaborate and reflect with other educators and take responsibility for transforming my practice.

Figura 2 Flipped Learning Network (FLN). (2014) *The Four Pillars of F-L-I-P™*

Tal como lo abordan tanto los docentes como los investigadores, este modelo aumenta la participación activa, el compromiso, la interacción y la autonomía en los estudiantes, que disfrutan de una educación más personalizada y la posibilidad de dedicar tanto tiempo como necesiten para comprender nuevos conceptos.

Este enfoque se vuelve especialmente interesante cuando se combina con un modelo basado en la investigación paso a paso. El método 5 Es ofrece a los estudiantes una manera de conectar las ideas científicas con sus experiencias y aplicar su aprendizaje. Estos son los 5 Es:

### **Participar (Engage)**

El profesor utiliza actividades cortas para promover la curiosidad. La actividad debe conectar conocimientos previos a nuevas experiencias de aprendizaje para exponer cualquier idea errónea y preparar a los estudiantes para un nuevo aprendizaje.

### **Explora**

Una investigación de laboratorio o actividades prácticas generalmente se introducen en esta fase a medida que los estudiantes intentan investigar un problema. Las ideas contradictorias, las preguntas y la confusión son comunes y ayudan a los estudiantes a identificar lo que necesitan saber antes de que se introduzcan nuevos términos o conceptos en la fase Explain.

### **Explicar**

Con la orientación del profesor, los estudiantes explican los conceptos que exploraron en la fase anterior y demuestran su comprensión de los nuevos términos que se introdujeron. Dependiendo del tema y el nivel de grado, la instrucción dirigida por maestros puede ser necesaria para abordar cualquier confusión y preguntas que surjan en la fase Explorar. Las preguntas pueden hacer que el aprendizaje sea más significativo, interactivo y participativo.

### **Elaborado**

Los estudiantes aplican sus conocimientos a nuevas experiencias y amplían su comprensión conceptual a medida que resuelven un problema en un nuevo contexto antes de la evaluación en la última fase del modelo 5E. Las actividades de elaboración pueden tener lugar durante el tiempo de clase, o pueden ser una tarea de tarea.

## **Evaluar**

Los estudiantes evalúan su aprendizaje y demuestran su comprensión y dominio de los conceptos clave. La evaluación no tiene que limitarse a una prueba o prueba. Puede ser un producto como una presentación, un póster, un folleto, un artículo de revista o un artículo final.

Entonces, ¿cuáles son los resultados de combinar estos dos enfoques? Goa y Foon Hew (2022) han llevado a cabo un estudio basado en el marco 5-E en un aula volteada cuestionando qué tan bien los estudiantes de primaria pueden entender los conceptos de PC. Probaron 125 estudiantes en el grupo experimental y 122 estudiantes en el grupo de control y resultó que los estudiantes que se unieron a la clase basado en el modelo de aula 5-E Flipped mejoraron significativamente su comprensión de los conceptos de PC y la resolución de problemas en comparación con el otro grupo que se unió a un enfoque de aula tradicional. Para ser más concretos, Goa y Foon Hew enfatizan en su estudio que el marco 5-E combinado con la metodología volteada ayuda a fomentar la participación activa de los estudiantes, por ejemplo: La fase de elaboración durante las actividades en clase ayuda a los estudiantes a encontrar soluciones por sí mismos. «Pensar en una tarea sin respuestas inmediatas del instructor puede conducir a un procesamiento cognitivo más profundo de los estudiantes, lo que a su vez puede ayudar a los estudiantes a comprender mejor el contenido» (Deslauriers et al., 2019 citado en Goa y Foon Hew, 2022). Además, el estado de que el grupo volteado era en general más activo y exploraba más soluciones a los problemas presentados.

Con todo, muestra que la combinación de estos dos modelos ayuda a los estudiantes a obtener una comprensión más profunda de los conceptos de PC y les ayuda a practicar activamente PC en clase y mejora su experiencia de aprendizaje.

### 3.2 Enfoques y estrategias pedagógicos actuales para la enseñanza y el aprendizaje en línea

La situación causada por el brote de COVID-19 obligó a los profesores a adaptarse al entorno de aprendizaje en línea. Más profesores que nunca tuvieron que buscar una manera de utilizar eficientemente las herramientas en línea para continuar con su trabajo. Muchos maestros no estaban preparados para esta situación y tuvieron que comenzar a enseñar sin las habilidades adecuadas.

Aunque la situación fue un desafío, también fue una oportunidad para que los profesores encuentren la gran cantidad de recursos en línea creados en los últimos años con el propósito de hacer que el proceso de enseñanza sea más fácil y entretenido.

La pandemia no solo obligó a los maestros a abandonar las aulas físicas, sino que también llamó la atención de los investigadores sobre conceptos relacionados con las competencias del siglo XXI como la alfabetización digital, la alfabetización TIC y el Pensamiento Computacional. Los profesores que se vieron obligados a superar las barreras causadas por la falta de experiencia en el entorno de enseñanza en línea ahora están más motivados que nunca para aprender cómo proporcionar una formación de buena calidad para los estudiantes con el uso de herramientas en línea.

Después de la revisión de la literatura, nos damos cuenta del importante papel del PC en el desarrollo futuro de los sistemas sociales, económicos y tecnológicos. El cambio es a menudo un proceso complejo con desafíos que superar. El proceso de cambio aumenta las oportunidades de errores, crea agitación en el estado psicosocial de las personas que se ven afectadas por esos cambios, pero sin cambio — el crecimiento es imposible.

En las últimas dos décadas, las escuelas, los centros de formación y las universidades han comenzado a abordar los desafíos y oportunidades creados por muchos recursos de aprendizaje en línea. El cambio del modelo

tradicional de aprendizaje presencial al aprendizaje a distancia se hizo cada vez más visible, o incluso imposible de ignorar.

Sin embargo, los profesores cuyas competencias docentes se desarrollaron en un aula tradicional pueden encontrar el cambio a la pedagogía en línea desalentadora. El aprendizaje en línea, especialmente el diseño de cursos completos en línea, requiere un esfuerzo coordinado de muchos jugadores involucrados. Es por eso que existe la necesidad de recursos didácticos que aborden este problema y proporcionen los conocimientos concretos necesarios para llevar a cabo el proceso de enseñanza en línea.

Rogers' Difusión de la teoría de la innovación (2003), subraya que la innovación se comunica a través de ciertos canales a lo largo del tiempo entre los miembros de un sistema social. Algunas innovaciones son exitosas, otras nunca son ampliamente aceptadas. Según Papert (1980) «las escuelas no pueden adelantarse a la sociedad y el desarrollo de una alfabetización digital requiere esencialmente tiempo».

Los profesores de hoy en día no solo tienen que adaptarse a los cambios dramáticos traídos por el auge de la web y las herramientas de comunicación móvil, sino que también necesitan satisfacer las necesidades de los estudiantes que utilizan Internet de manera regular y efectiva (Johnson et al. 2012). Para digitalizar su aula, los profesores deben incorporar la mejor pedagogía tradicional en el aula para abordar las necesidades de los nuevos entornos de aprendizaje en línea.

Los sistemas sociales complejos se benefician de una estrategia clara de implementación de cambios, y el PC puede ofrecer un método necesario para desarrollar dicha estrategia. Según Wing (2006) PC implica «tomar un enfoque para resolver problemas, diseñar sistemas y comprender el comportamiento humano que se basa en conceptos fundamentales para la computación». PC ofrece una construcción para resolver problemas relacionados con el cambio social a través del uso de algoritmos, patrones,



jueguen con modelos interactivos (Beheshti, 2017). Las estrategias de enseñanza de PC se describirán en la siguiente sección de este capítulo.

### 3.2.1 Herramientas de enseñanza en línea

A través de este subcapítulo, nos estamos centrando en los enfoques pedagógicos actuales para la enseñanza del Pensamiento Computacional en línea. Después de la revisión de la literatura, categorizamos las siguientes categorías de tipos de herramientas: lenguajes de programación visuales y de otro tipo, videojuegos y herramientas de colaboración.

#### **Uso del lenguaje de programación visual (LPV) en el proceso de aprendizaje por PC**

En computación, un lenguaje de programación visual (sistema de programación visual, LPV) es cualquier lenguaje de programación que proporciona elementos gráficos o icónicos que pueden ser manipulados por los usuarios de una manera interactiva. En otras palabras, cualquier lenguaje que permita a los usuarios crear programas manipulando los elementos del programa gráficamente en lugar de especificarlos textualmente. Debido a que los lenguajes LPV no requieren un conocimiento específico de codificación, son el enfoque más popular para enseñar programación y habilidades de PC en la escuela. Escribir código es en general un procedimiento muy creativo y muchos estudios han demostrado que los estudiantes que experimentaron la escritura de código mejoran sus habilidades de resolución de problemas y habilidades de diseño de sistemas (Krugel, Hubwieser, 2017).

Un LPV permite la programación con expresiones visuales, arreglos espaciales de texto y símbolos gráficos, utilizados como elementos de sintaxis o notación secundaria. Por ejemplo, muchos LPV (conocidos como flujo de datos o programación de diagramas) se basan en la idea de «cajas y flechas», donde las cajas u otros objetos de pantalla se tratan como entidades, conectadas por flechas, líneas o arcos que representan relaciones.

Sin embargo, al usar un lenguaje de programación visual, debemos pensar en cómo presentarlo a los estudiantes para que puedan usarlo de manera efectiva y, sobre todo, querer desarrollar sus habilidades en su uso.

### **Esfuerzo individual**

Un método es animar a los estudiantes a trabajar individualmente. En este caso, al estudiante se le da una tarea para realizar o un problema para resolver y trabaja en ella. Por ejemplo, Ahmadi y Jazayeri (2014), trataron de ver si un novato completo podía aprender de forma independiente habilidades de PC pidiendo a los participantes de su estudio que exploraran el problema propuesto y luego obtuvieran las habilidades de programación que se requieren para resolverlo. En consecuencia, el 90 % de las tareas fueron completadas por los estudiantes participantes.

### **Colaboración y remezcla**

El remezcla se ha definido como la reelaboración y combinación de artefactos creativos existentes, generalmente en forma de música, video y otros medios interactivos. Lessig (2008) ha sugerido que la remezcla refleja un amplio cambio cultural impulsado por Internet y una fuente de enorme potencial creativo. Manovich (2020) ha llamado el remezcla “una característica incorporada del universo de los medios digitales en red.

La importancia de la remezcla y reutilización ha sido reconocida en la esfera del desarrollo de habilidades de Pensamiento Computacional. Los estudiantes, además de crear su propio código, también pueden remezclar un código creado por otra persona. En el contexto del PC:

- **Reutilizar** significa tomar pedazos de código creados por otros y usarlo para resolver un problema o satisfacer una necesidad, en lugar de crearlo desde cero.
- **El remezcla** implica juntar o «mashing up» código para video, sonido, texto, etc. creado por otras personas para hacer algo nuevo y original.

Un ejemplo es un proyecto Scratch que ha sido creado por un par de personas, y luego pasó a otro par para extender y reimaginar.

Otro ejemplo es dado por Repenning et al. (2009). Los autores de este estudio creían que la programación se enseña hoy de manera ineficiente y tiende a desalentar la interacción entre pares de los estudiantes. En su estudio emplearon un sistema de envío de tareas en línea llamado Scalable Game Design Arcade (SGDA). SGDA proporcionó una forma complementaria de enseñar Pensamiento Computacional utilizando interacciones entre pares. Los estudiantes aprendieron unos de otros, viendo y ejecutando el código del otro. Se hizo con el fin de implementar los Principios de Flujo de Inspiración en Diseño Educativo de Juegos.

El método de colaboración que promovieron se centró en los Principios de Flujo de Inspiración, que fueron el resultado de 10 años de observación de estudiantes de secundaria en clubes de informática que compartían libremente ideas de programación, códigos y consejos sobre soluciones de resolución de problemas. La intención de los investigadores era transferir esas interacciones presentes en el entorno del club de computación al aula. Los principios, que ellos nombraron, incluyen:

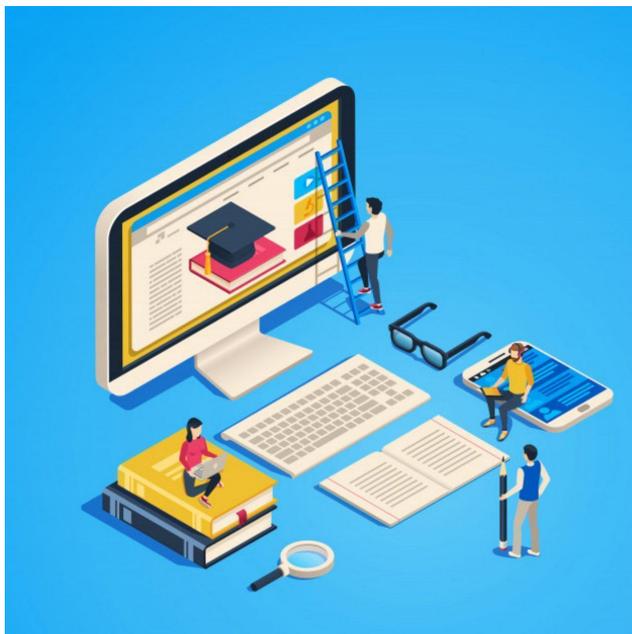
- Mostrar proyectos en un foro público
- Visualización y ejecución de proyectos de compañeros de estudios
- Proporcionar comentarios sobre los proyectos de otros estudiantes
- Descargar y ver código para cualquier proyecto
- Proporcionar motivación para que los estudiantes vean, descarguen y envíen comentarios sobre los proyectos de sus compañeros de clase

La Arcade de Diseño de Juego escalable (SGDA) que utilizaron se basó en la idea de Diseño de Juego escalable. Este enfoque combina marcos de motivación y competencia, estándares para la tecnología y herramientas de creación de pensamiento computacional. Muestra que diseñar y jugar videojuegos puede ser un método de enseñanza de los principios de PC

(Wing, 2006; Ioannidou et al., 2011). El currículo de Diseño de Juego escalable equilibra los desafíos de diseño y las habilidades de diseño para mantener a los estudiantes en un flujo óptimo. Esto se puede lograr a través de varias formas de métodos de andamio, como la instrucción explícita justo a tiempo y el apoyo social de las interacciones con los instructores y otros estudiantes.

Ahora que hemos explicado lo que es LPV, describimos los métodos de enseñanza de PC basados en resolver problemas de forma independiente o colaborar y remezclar con la interacción estresada entre pares, ahora es el momento de seguir adelante y hablar sobre las herramientas que un maestro puede usar para enseñar PC.

### **Herramientas de lenguaje de programación visual**



# **Online Teaching Tools**

Ioannidou et al., (2011) afirman que las Herramientas de Autorización de Pensamiento Computacional son el tipo de herramientas que son esenciales para permitir que los niños adquieran habilidades de Pensamiento

Computacional. Herramientas de creación de PC que son de alta calidad y permiten el aprendizaje de PC:

tener un umbral bajo: un estudiante puede producir un juego de trabajo rápidamente.

tienen techo alto: un estudiante puede hacer un juego real que es jugable y exhibe un comportamiento sofisticado, por ejemplo, IA compleja.

tener flujo de andamios: la herramienta + currículo proporciona escalones con habilidades gestionadas y desafíos para acompañar a la herramienta.

habilitar la transferencia: Tool + curriculum debe trabajar tanto para el diseño de juegos como para las aplicaciones de ciencias computacionales posteriores, así como para la transferencia de apoyo entre ellas. También debería facilitar la transición a lenguajes de programación profesionales.

apoyo a la equidad: las actividades de diseño de juegos deben ser accesibles y motivacionales a través de los límites de género y etnia.

son sistémicos y sostenibles: la combinación de la herramienta y el plan de estudios puede ser utilizada por todos los profesores para enseñar a todos los estudiantes (por ejemplo, incluir oportunidades para la formación del profesorado y el apoyo a la implementación; alinearse con los estándares y trabajar en áreas de contenido STEM).

La siguiente tabla contiene una lista de herramientas de PC notables. Muchos de los cuales permiten el diseño del juego:

<b>Nombre</b>	<b>Descripción</b>	<b>Nombre del programa/Autores, desarrolladores/Licencia</b>
---------------	--------------------	--

<b>Kodu</b>	<p>— permite desarrollar juegos 3D de un tipo diferente: por ejemplo, aventura, arcade, carreras</p> <p>— para niños de 9 a 10 años</p>	Licencia de código abierto de Microsoft FUSE Labs/Microsoft
<b>Greenfoot</b>	<p>— es un entorno de desarrollo integrado utilizando Java o Stride</p> <p>— permite desarrollar aplicaciones gráficas bidimensionales, como simulaciones y juegos interactivos</p>	M. Kölling/Libre y código abierto
<b>NetLogo</b>	<p>— es un entorno de modelado programable para simular fenómenos naturales y sociales</p> <p>— los estudiantes pueden abrir simulaciones y «jugar» con ellos, explorando su comportamiento en diversas condiciones</p> <p>— permite a los estudiantes, profesores y desarrolladores de currículos crear sus propios modelos.</p>	U. Wilensky, Universidad del Noroeste/Libre y código abierto
<b>Scratch</b>	<p>— permite desarrollar historias interactivas, animaciones y simulaciones</p> <p>— desarrollo de juegos 2D de un tipo diferente: clicker juegos, juegos de plataforma, juego de laberinto, juego de pong</p> <p>— los juegos desarrollados con Scratch se pueden jugar localmente o en línea una vez que los juegos se han subido en el sitio web de Scratch</p>	MIT Media Lab/Libre y de código abierto

	— recomendado para edades de 8 a 16 años	
<b>ScratchJr</b>	— es una interpretación de Scratch diseñada principalmente para audiencias más jóvenes (5-7 años).	MIT Media Lab/Libre y de código abierto
<b>AgentCubes</b>	— es un lenguaje de programación educativo  — permite crear juegos y simulaciones en línea 3D y 2D  — recomendado para las edades 10-12	A. Repensión/licencia de propiedad
<b>Snap</b>	— Interfaz de D&D  — Snap te permite crear juegos, animaciones, apps, presentaciones, etc.  — recomendado para personas de 12 a 20 años	B. Harvey y J. Mönig/Libre y código abierto
<b>App Inventor</b>	— utiliza un lenguaje visual, bloquea  — permite la creación de aplicaciones Android  — recomendado para edades de 9-12 años	H. Abelson, M. Friedman, MIT Media Lab/Libre y código abierto
<b>VTS Editor</b>	— VTS Editor permite a los usuarios desarrollar juegos de simulación	Grave de fábrica/Pagado con prueba gratuita
<b>AgentSheets</b>	— entorno de programación basado en bloques  — permite desarrollar juegos 3D y publicarlos en la web  — para niños, edad no especificada	A. Repensión/licencia de propiedad

<b>Alice</b>	<ul style="list-style-type: none"> <li>— entorno de programación basado en bloques</li> <li>— permite desarrollar juegos 3D y publicarlos en la web</li> <li>— recomendado para edades de 13 a 18 años</li> </ul>	Universidad del Sarre/LicenciaMIT
<b>GameMaker</b>	<ul style="list-style-type: none"> <li>— utiliza GameMaker Language</li> <li>— Interfaz de D&amp;D</li> <li>— permite desarrollar juegos 2D de un tipo diferente, por ejemplo: dispararles juegos</li> <li>— recomendado para edades de 11 a 14 años</li> </ul>	M. Overmars, Yoyo Juegos/Licencia de propiedad
<b>Bubble</b>	<ul style="list-style-type: none"> <li>— se utiliza para crear sitios web y aplicaciones web</li> </ul>	J. Haas E. Straschnov/Comercial
<b>GDevelop</b>	<ul style="list-style-type: none"> <li>— usa JavaScript</li> <li>— Motor de desarrollo de juegos 2D</li> <li>— el usuario puede desarrollar todo tipo de juegos, por ejemplo, juegos de plataforma, rompecabezas, juegos de disparos de balas</li> <li>— recomendado para edades de 7 a 19 años</li> </ul>	F. Rival, V. Levasseur, A. Vivet, A. Pacaud, Franco M., T. Imreorov/Free and open-source
<b>Godot</b>	<ul style="list-style-type: none"> <li>— utiliza GDScript, C#, VisualScript y C++ y C</li> <li>— un motor increíblemente versátil para juegos 2d y 3D</li> </ul>	J. Linietsky, A. Manzur/Libre y código abierto

<b>Celestory</b>	<ul style="list-style-type: none"> <li>— Interfaz de D&amp;D</li> <li>— permite desarrollar juegos 2D de un tipo diferente: Juego de cartas, juego de escape, plataforma, interactivo, película, cuestionario</li> </ul>	Celestory 2022/freemium
<b>Catrobat</b>	<ul style="list-style-type: none"> <li>— un lenguaje de programación visual basado en bloques</li> <li>— permite la creación de juegos, historias, animaciones y muchos tipos de otras aplicaciones directamente en teléfonos inteligentes.</li> </ul>	W. Slany/Libre y de código abierto
<b>GameSalad Creator</b>	<ul style="list-style-type: none"> <li>— Interfaz de D&amp;D</li> <li>— permite desarrollar juegos en 2D de un tipo diferente como puzles o arcade</li> <li>—permite al usuario crear aplicaciones para iOS, Android, HTML5 y la Plataforma Mac</li> <li>— recomendado para mayores de 12 años</li> </ul>	GameSalad/Licencia de propiedad
<b>GameStar Mechanic</b>	<ul style="list-style-type: none"> <li>— apuntar y hacer clic en la interfaz</li> <li>— permite desarrollar juegos 2D de un tipo diferente: aventura, plataforma, acción y experimental</li> <li>— desarrollado juego subido en el Game Alley y compartido con la comunidad en línea</li> <li>— diseñado especialmente para las edades de 14 a 17 años</li> </ul>	E-Line Media/Libre y de código abierto

<b>Stagecat Creator</b>	<ul style="list-style-type: none"> <li>— apuntar y hacer clic en la interfaz.</li> <li>— permite desarrollar juegos 2D de un tipo diferente: juegos de acción y aventura.</li> <li>— permite a los usuarios guardar los juegos desarrollados en un disco local o en Internet.</li> <li>— diseñado para niños de tan solo 8 años</li> </ul>	A. Cypher y D. Canfield Smith, ATG/Freeware
<b>Unity</b>	<ul style="list-style-type: none"> <li>— utiliza el lenguaje C#</li> <li>— permite a los usuarios diseñar juegos en 2D y 3D</li> <li>—se pueden crear diferentes tipos de juegos: Primera persona, Volar, Puzzle, Rolling</li> </ul>	Unity Technologies/Licencia de propiedad
<b>Blender Game Engine</b>	<ul style="list-style-type: none"> <li>— Python como lenguaje de programación</li> <li>— Suite de creación 3D</li> </ul>	Blender Foundation/Libre y de código abierto
<b>Babylon.js</b>	<ul style="list-style-type: none"> <li>— utiliza JavaScript; TypeScript</li> <li>— es un motor 3D para juegos y otras visualizaciones 3D. Permite construir gráficos animados en 3D</li> </ul>	D. Catuhe, Microsoft y colaboradores/Libre y código abierto
<b>Construct/ Construct 2</b>	<ul style="list-style-type: none"> <li>— Basado en HTML5</li> <li>— permite desarrollar juegos en 2D</li> </ul>	Scirra/licencia de propiedad
<b>RPG Maker VX Ace Lite</b>	<ul style="list-style-type: none"> <li>— utiliza el lenguaje Ruby</li> <li>— es un motor de juego diseñado para hacer juegos de rol en 2D</li> </ul>	Enterbrain Inc./Versión gratuita y de pago

<b>Flowgorithm</b>	<ul style="list-style-type: none"> <li>— utiliza el lenguaje C#</li> <li>— herramienta que permite a los usuarios escribir y ejecutar programas utilizando diagramas de flujo</li> <li>— puede traducir interactivamente programas de diagramas de flujo en código fuente escrito en otros lenguajes de programación</li> </ul>	D. Cocinar/Freeware
<b>Hopscotch</b>	<ul style="list-style-type: none"> <li>— interfaz de arrastrar y soltar</li> <li>— permite crear juegos, animaciones y otros coloridos programas interactivos</li> <li>— diseñado para las edades 10-16</li> </ul>	Hopscotch Technologies 2021/Aplicación gratuita para descargar. Versión actualizada de pago.
<b>Kojo</b>	<ul style="list-style-type: none"> <li>— una interfaz de apuntar y hacer clic</li> <li>— permite crear dibujos, animaciones, juegos</li> <li>— diseñado para niños de tan solo 8 años</li> </ul>	Lalit Pant/Free y de código abierto
<b>Mblock</b>	<ul style="list-style-type: none"> <li>— basado en el lenguaje de programación visual del MIT</li> <li>— arrastrar y soltar la interfaz</li> <li>— permite crear juegos y animaciones</li> <li>— diseñado para niños de tan solo 8 años</li> </ul>	No lo sé. Wang/Libre y de código abierto
<b>Open Roberta</b>	<ul style="list-style-type: none"> <li>— permite a niños y adolescentes programar robots</li> <li>— diseñado para niños de tan solo 8 años</li> </ul>	B. Jost, R. Budde, T. Leimbach, A. Kapusta. Fraunhofer IAIS/Libre y de código abierto

<b>Raptor</b>	— diseñado específicamente para ayudar a los estudiantes a visualizar sus algoritmos	M. C. Carlisle, T. Wilson, J. Humphries y Jason Moore/Libre y código abierto
<b>StarLogo</b>	— utilizar gráficos 3D para hacer juegos y modelos de simulación	M. Resnick, E. Klopfer, D. Wendel, MIT/Libre de uso no comercial
<b>ToonTalk</b>	— permite construir programas y juegos — diseñado para niños de tan solo 5 o 6 años	K. Kahn/Versions 1.0 y 2.0 tenían licencias comerciales versión 3.0 ahora es gratis.
<b>Visual Logic</b>	— permite a los estudiantes escribir y ejecutar programas utilizando diagramas de flujo	© PGS Systems/Licencia de propiedad

### Jugar a videojuegos

El uso de LPV es un buen método de enseñanza de PC porque permite a los estudiantes construir artefactos como videojuegos o historias interactivas. Pero también tenemos otros métodos que son dignos de consideración (Weintrop et al. 2016).

Muchos estudios demuestran cómo simplemente jugar un videojuego puede ser una forma efectiva de practicar PC. Este método puede ser especialmente atractivo para experiencias de aprendizaje fuera del aula. El método correcto para enseñar PC siempre estará determinado por una variedad de factores, es decir, la edad de los estudiantes, las preferencias de los maestros, las limitaciones de tiempo, etc.

La mayoría de los juegos están diseñados con una actividad «constructionista» como núcleo de la jugabilidad. La otra teoría a tener en cuenta es Computational Encoding (Holbert & Wilensky, 2011), y también hay varias técnicas de aprendizaje de PC como remezcla de colaboración e

inmersión (Pellas y Peroutseas 2016; Dhatsuwan y Precharattana 2016; Debabi & Bensebaa, 2016).

A continuación, se muestra una lista de los videojuegos más conocidos y gratuitos que ayudan de manera transparente a los usuarios a mejorar los principios de PC.

- **AlgoGame:** Futschek (2006), propuso el AlgoGame como método de enseñanza de PC. Se pidió a los estudiantes que se centraran en resolver problemas en lugar de aprender la sintaxis de un lenguaje de programación. Futschek informó que después de tocar el AlgoGame, un grupo experimental tuvo mejores resultados al escribir un algoritmo de selección que un grupo de control.
- **CodeCombat:** El juego llamado CodeCombat ofrece una plataforma ideal para enseñar conceptos de programación orientada a objetos (OOP). Enseñar OOP ayuda a preparar a los estudiantes para escribir su primer programa mediante el uso de clases y objetos. CodeCombat tiene múltiples niveles, y cada nivel cubre varios conceptos del currículo de ciencias de la computación. Por ejemplo, el primer nivel cubre conceptos introductorios como sintaxis, secuencia, objetos y métodos. El juego ilustra objetos como bloques de construcción, son cosas o personajes que pueden realizar acciones. En el juego, un héroe es un objeto. Las acciones que realiza un objeto se llaman métodos. Esta es la forma en que los estudiantes tienen que interactuar con los bloques de codificación (Karram, 2021).
- **Rob-Bot:** es un juego de cartas de pensamiento computacional que consta de un paquete de 57 cartas, cada una de las cuales contiene 8 personajes u objetos diferentes. Cada carta contiene un carácter u objeto que coincide con uno en cada otra carta del paquete. Los autores afirman que el juego desarrolla el pensamiento CT porque requiere identificar personajes y objetos específicos en las cartas (descomposición), buscar y encontrar el mismo personaje u objeto

(reconocimiento de patrones), abstracción a través de descartar y filtrar personajes y objetos que no coinciden y, finalmente, diseño de algoritmos, es decir, una técnica de búsqueda e identificación de personajes y objetos (Rob-Bot Resources, 2018).

### 3.2.2 PC en la educación en línea

Hasta ahora, hemos presentado un enfoque pedagógico consistente y completo para el uso de métodos y herramientas para la educación en línea. Esta sección está dedicada al papel de los métodos de enseñanza en línea con el fin de impulsar y apoyar el desarrollo del Pensamiento Computacional.



Según Voskoglou y Buckley (2012), *«PC desarrolla una variedad de habilidades (lógica, creatividad, pensamiento algorítmico, modelado/simulaciones), implica el uso de metodologías científicas y ayuda a desarrollar tanto la inventiva como el pensamiento innovador»*.

El pensamiento computacional ofrece a los profesores no solo la oportunidad de explorar nuevos procesos de resolución de problemas, sino también de dar rienda suelta a su propia creatividad en el diseño de cursos, investigaciones y políticas dentro de sus áreas de especialización. Una forma de ofrecer a los

profesores la oportunidad de aprender Pensamiento Computacional sería proporcionar una oportunidad para emprender el desarrollo de cursos en línea, utilizando el Pensamiento Computacional como modelo. Los profesores podrán elegir cualquier tema de interés; el diseño participativo y el desarrollo del contenido de los cursos en línea promoverían la comprensión de la nueva tecnología del curso y permitirían a los profesores aplicar su experiencia en áreas de contenido del curso.

Los profesores pueden encontrar uno de los modelos en Soh et al. (2015), quien creó una serie de cursos que apoyaron tanto el análisis (descomposición, reconocimiento de patrones) como la reflexión (abstracción, diseño de algoritmos), y utilizó ejercicios de descripción simples para promover la resolución colaborativa de problemas. Smidt et al. (2014) también ofreció un posible marco para el desarrollo, comenzando con una conversación cara a cara y luego transfiriendo comunicaciones a un foro de discusión en línea. Masterman, Walker, & Bower, (2013) tenía como objetivo promover oportunidades en el uso de herramientas tecnológicas para guiar y apoyar la pedagogía, al tiempo que promovía el aprendizaje de profesores colaborativos en el diseño de cursos en línea. Concluyeron que *«[el] desafío de integrar el apoyo computacional para el pensamiento de los maestros de una manera que tenga en cuenta todos estos factores es posicionarlo dentro del diseño de un programa, departamento y facultad donde es utilizado por la escuela de forma regular»*.

En la práctica, hay dos barreras para integrar el Pensamiento Computacional en el currículo

- la falta de consenso sobre una definición de Pensamiento Computacional;
- la escasez de maestros cualificados que puedan enseñar esta habilidad (Chi y Menekse 2015; Curzon y Dorling 2014).

Esta falta de acuerdo se destacó en 2009 cuando un taller organizado por el Consejo Nacional de Investigación de Estados Unidos con el objetivo de establecer «*El alcance y la naturaleza del pensamiento computacional*» no logró llegar a un consenso entre sus participantes sobre el contenido y la estructura del Pensamiento Computacional (Committee for the Workshops on Computational Thinking et al. 2010). El pensamiento computacional tiene su comienzo con Seymour Papert (1980) y su muy citado libro “*Mindstorms: Niños, Computadoras e Ideas Poderosas*” pero fue Jeannette Wing (2006), quien popularizó este concepto. Ella lo definió de la siguiente manera: «*El pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, recurriendo a los conceptos fundamentales para la informática*». Esta definición de Wing tiene importancia para la educación obligatoria, ya que afirma que el Pensamiento Computacional es fundamentalmente un proceso de pensamiento, es decir, independiente de la tecnología y que sus soluciones pueden ser ejecutadas por un ser humano o un ordenador, o ambos (Comisión Europea. Centro Común de Investigación. 2016).

El artículo de Wing de 2006 fue muy específico sobre esto afirmando que el Pensamiento Computacional se trataba de conceptualización, no de programación. Se refería a las ideas, no a los artefactos.

Sin embargo, esta no es una creencia universal. Por ejemplo, (Brennan y Resnick (2012) propusieron una definición de Pensamiento Computacional que giraba en torno al lenguaje Scratch. En el taller mencionado, organizado por el Consejo Nacional de Investigación de los Estados Unidos en 2009 (Brennan y Resnick 2012; Wolz et al. 2010) todas las opiniones expresadas coinciden en que la programación es esencial para el Pensamiento Computacional (Comité de los Talleres de Pensamiento Computacional et al. 2010).

La discusión anterior destaca que no hay una respuesta simple a la pregunta de qué es el Pensamiento Computacional, ilustrando así uno de los desafíos

para integrar el Pensamiento Computacional en el aula y en el entorno online.

### 3.3 Pensamiento computacional: Métodos de enseñanza

#### 3.3.1 Formación en programación y computación matemática como método de enseñanza del Pensamiento Computacional

El pensamiento es el reflejo indirecto y abstracto de un cerebro humano de la esencia de los objetos, sus conexiones internas y también es la forma principal de las actividades intelectuales humanas. Cómo expresar explícitamente las actividades de pensamiento internalizadas en el cerebro humano y hacer que los estudiantes vean y toquen el conocimiento abstracto de la computadora con propiedades no físicas es una pregunta muy importante.

El PC es un método de pensamiento y mientras enseña el objetivo básico de los maestros de PC debe ser desarrollar las habilidades de pensamiento de los estudiantes para resolver problemas reales mediante la aplicación consciente de PC. Edsger Dijkstra, uno de los famosos expertos en computación que publicó el libro «La Enseñanza de la Programación, es decir, la Enseñanza del Pensamiento» en 1976, dijo: *«las herramientas que utilizamos habían afectado las formas y hábitos de nuestro pensamiento, por lo que también afectarán profundamente nuestra capacidad de pensamiento»*. Por lo tanto, el PC y la programación son conceptos estrechamente relacionados. PC proporciona algunos métodos de pensamiento eficientes para los cursos de programación y los cursos de programación proporcionan un soporte práctico para el cultivo de habilidades de PC. El objetivo del curso de programación es hacer que los estudiantes entiendan cómo resolver problemas reales con el uso de computadoras, que es la encarnación de la formación de PC. PC se implementa mediante cálculo automático y mecánico y el contenido de un curso de programación o matemática puede incluir, por ejemplo: abstracción de problemas, descomposición de problemas, creación de modelos, diseño de algoritmos y verificación de resultados.

Gracias al método de enseñanza basado en problemas, las actividades de enseñanza incluyen encontrar un problema real, mapear métodos de resolución en un modelo informático y diseñar algoritmos.

En los cursos de programación tradicionales, a los estudiantes se les enseña cómo programar con un idioma específico. Los métodos aplicados carecen de universalidad, la capacidad de programar en un idioma no se transfiere a ser capaz de programar en otro. En consecuencia, los efectos de la enseñanza fueron muy diferentes de los objetivos educativos

Como sabemos, los lenguajes de programación son diferentes entre sí, pero los métodos de resolución de problemas y pensamiento son los mismos o similares. La introducción de PC obliga a los estudiantes a prestar más atención a las ideas detrás de la enseñanza y los métodos de pensamiento.

Los métodos de enseñanza basados en PC pueden ayudar a los estudiantes a comprender la relación de colaboración entre el pensamiento humano y la computación, también pueden hacer que dominen los métodos generales de pensamiento resumidos por PC. PC cultiva las habilidades de los estudiantes para encontrar problemas, analizar problemas, diseñar algoritmos, comparar el rendimiento y resolver problemas mediante la aplicación de métodos y principios específicos.

Los estudiantes pueden tomar lo que aprendieron durante el curso de programación y aplicar habilidades de pensamiento computacional a diferentes áreas de la vida. Por lo tanto, un curso de programación puede reflejar la mayoría y mejorar el proceso de pensamiento de PC.

1. ¿Es la programación la forma más adecuada para expresar PC?

PC se describe a través de algunos idiomas o palabras. Y no tiene sentido sin expresión explícita. Además, la expresión de PC debe seguir algunas reglas estrictas y formatos de lenguaje específicos, de lo contrario, no se puede entender. El lenguaje de programación en sí es una especie de expresión

formal con las características de determinación, finitud y mecanización. El lenguaje de programación puede expresar PC con precisión y la expresión puede ser entendida por computadora fácilmente.

2. ¿La enseñanza de programación incluye los métodos de pensamiento de PC?

El curso de programación convierte los problemas desconocidos en los problemas conocidos mediante la aplicación de los métodos de reducción, transformación, simulación, etc. El curso de programación simplifica los problemas complejos en simples mediante la aplicación de la programación estructural y las funciones basadas en el enfoque de separación de preocupaciones del PC. El curso de programación resuelve los problemas de incertidumbre basados en el enfoque heurístico del PC. El curso de programación mejora la velocidad de cálculo mediante el enfoque paralelo de PC y la evaluación del tiempo y el espacio propuesto por PC es también el índice importante para medir el rendimiento de los algoritmos; El curso de programación se centra en la robustez y confiabilidad y los algoritmos se pueden recuperar en el peor de los casos a través de los métodos de prevención, protección, redundancia, tolerantes a fallas, contención de daños y corrección de errores de PC, etc.

3. ¿Es la práctica en un curso de programación una manera importante para entrenar la capacidad de PC?

El cultivo de habilidades de PC requiere experiencia. Los estudiantes deben ser capaces de probar los conceptos que aprendieron y ver si pueden aplicarlos en la práctica. En el caso del aprendizaje por PC, la experiencia puede ser en forma de escribir un programa de computadora o diseñar un juego.

A través de mucha práctica, los cursos de programación pueden transformar el conocimiento superficial y abstracto en la aplicación en la práctica de los métodos PC apropiados y el conocimiento informático efectivo. En

conclusión, con la aplicación de PC, el conocimiento de programación corrió a lo largo de todo el proceso de resolución de problemas para el objetivo final de formar una red de pensamiento complicada, que tomó el conocimiento como nodos y el PC como conectores. PC hace que el contenido del curso de programación se acerque al problema real en la medida de lo posible y expanda el pensamiento humano y el conocimiento de la computadora desde la computadora al mundo real al romper las limitaciones entre los seres humanos, el mundo real y la informática. El PC y la programación tienen una relación natural, lo que indica que el cultivo de la capacidad de PC necesita el apoyo de la programación y la programación necesita la orientación del PC. Por lo tanto, tiene sentido introducir PC en la programación. Era necesario enseñar el conocimiento junto con enseñar su pensamiento contenido. De esta manera, puede fortalecer el conocimiento a través del pensamiento y puede cultivar el pensamiento a través del conocimiento. En última instancia, puede ayudar a los estudiantes a aprender lenguajes de programación y cultivar su capacidad de PC.

### **3.3.2 Aprendizaje experiencial: aprender haciendo en la educación PC**

Esta sección presentará diferentes enfoques del aprendizaje experiencial como métodos de enseñanza para el uso del Pensamiento Computacional en un aula en línea. El objetivo es presentar los métodos que caen bajo el enfoque de aprendizaje experiencial en el contexto del enfoque del pensamiento computacional y proporcionar orientación práctica para la enseñanza en línea.

El aprendizaje experiencial se define a continuación. A continuación, se presentan los diferentes métodos que entran en él: Basado en proyectos, basado en problemas, aprendizaje cooperativo y aprendizaje basado en juegos. La atención se centra en su conexión con el PC y la implementación de esta combinación en la enseñanza en línea.

¿Qué es el aprendizaje experiencial (AE)? ¿Cómo se puede definir?

«La experiencia [aprendizaje] es una filosofía y metodología en la que los educadores se involucran intencionalmente con los estudiantes en experiencia directa y reflexión enfocada con el fin de aumentar el conocimiento, desarrollar habilidades y aclarar valores» (Asociación para la Educación Experimental, párr. 1).

En términos generales, AE es un proceso que describe el aprendizaje a través de la experiencia. Se trata de un aprendizaje práctico. Esto significa que AE tiene una perspectiva diferente sobre el proceso de aprendizaje en comparación con los enfoques didácticos y otras teorías de aprendizaje. Al situar la experiencia en el centro del aprendizaje (Kolb, 2014), AE se caracteriza por ciertas características que son contrarias a los métodos de aprendizaje tradicionales.

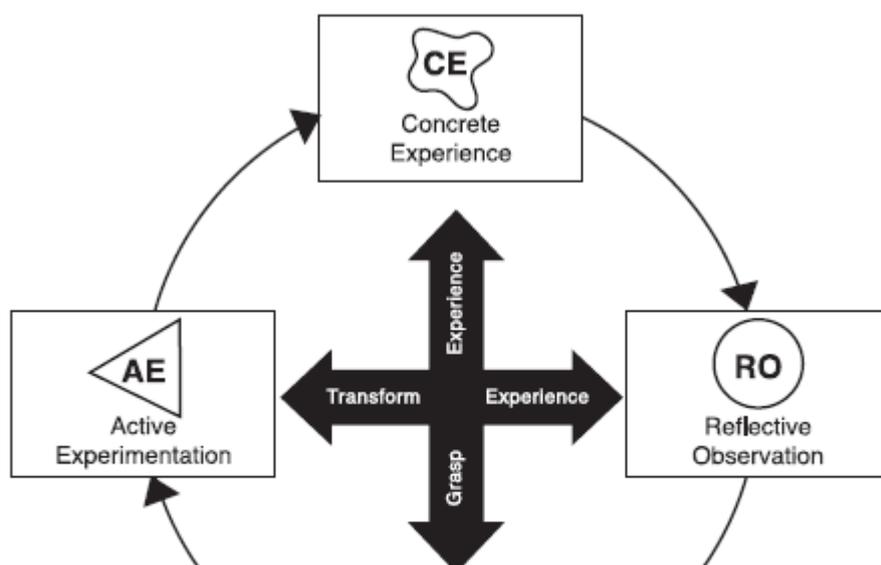
Dentro de las teorías de AE hay diferentes enfoques. Diferentes autores han atribuido un papel central en el aprendizaje a la experiencia. Lo que Kolb (2014) llama a los «estudiosos fundacionales del aprendizaje experiencial» son los siguientes autores: Wiliam James, John Dewey, Kurt Lewin, jean Piaget, Lev Vygotsky, Carl Jung, Mary Parker Follett, Carl Rogers y Paulo Freire.

Todos ellos han producido diferentes énfasis para la teoría. Sin embargo, han sido los psicólogos educativos John Dewey (1859-1952), Carl Rogers (1902-1987) y David Kolb (b. 1939) los que sentaron las bases de las teorías de AE con el enfoque en «aprender a través de la experiencia o «aprender haciendo». En general, AE es una teoría dinámica, holística y multidimensional. Se dirige a los estudiantes en su totalidad y son vistos como recursos valiosos en el ciclo educativo (Carver, 1996). Esto significa que son factores importantes para su propio viaje educativo, así como para uno de los otros estudiantes. Además, el personal educativo también es visto como un miembro de esta comunidad de aprendizaje. Durante el proceso, el maestro aprende constantemente de sus estudiantes (Carver, 1996). Esto significa que AE es un enfoque centrado en el alumno. Debido a que esa

experiencia es el centro del proceso de aprendizaje, se interpone la forma tradicional de instrucción. AE aborda instrucciones de diseño para involucrar a los estudiantes con el tema a través de experiencias directas y prácticas. donde la instrucción está altamente estructurada, los estudiantes en situaciones de aprendizaje experiencial cooperan y aprenden unos de otros en un enfoque más semiestructurado con el objetivo de encontrar soluciones para problemas del mundo real. Además, el papel del profesor cambia en el enfoque AE. El maestro es visto como un facilitador y guía y no solo la fuente de información primaria que da instrucciones directas donde los estudiantes solo la tarea es escuchar y absorber la información mientras el maestro está explicando (Northern Illinois University, 2012).

El enfoque ofrece numerosas ventajas. Entre otras cosas, sus defensores argumentan que puede conducir a una mejor y más profunda comprensión del curso y los conceptos. Los estudiantes aprenden mejor cuando aplican y experimentan los conceptos ellos mismos. Por lo tanto, no es la palabra reproducida la que se almacena en la memoria a corto plazo, sino que la información se almacena en la memoria autobiográfica y allí se almacena de manera más sostenible. A través del proceso descrito a continuación, los estudiantes aprenden a reflexionar críticamente (Slavich & Zimbardo, 2021).

Para aplicar el enfoque en el aula, los maestros pueden usar el concepto de Kolb del «ciclo de aprendizaje». El ciclo se basa en cuatro etapas que dividen la experiencia de aprendizaje en los siguientes pasos: acción, reflexión, experiencia y abstracción.



Esto significa que el ciclo de aprendizaje de Kolb combina la comprensión de la experiencia — Experiencia concreta (CE) y conceptualización abstracta (AC) — con la transformación de estas experiencias en conocimiento a través de la observación reflexiva (RO) y la experimentación activa (AE).

Además, el enfoque experiencial cambia la interacción entre estudiantes y profesores. Utilizando el ciclo de Kolb todos los participantes están incluidos en la experiencia de aprendizaje, los estudiantes y los profesores son tanto los receptores de la información como los creadores de la información (Kolb, 2017). Es un enfoque centrado en el estudiante y flexible.

¿Cómo se puede combinar AE con PC? Mirando los enfoques teóricos fundamentales, se puede ver cómo estas metodologías pueden trabajar de la mano.

El enfoque pedagógico principal que guía el pensamiento computacional es el construccionismo. Piaget sostiene que el aprendizaje es un proceso interno, que tiene lugar a través de la interacción con el entorno (Piaget, 1969). Por lo tanto, en lugar de ver o escuchar, pensar y reflexionar mejora nuestra capacidad de aprender. En consecuencia, las metodologías de aprendizaje activo son una herramienta para desarrollar el pensamiento computacional (ya sea una o una combinación de varias). Seymour Papert, la fuerza

impulsora detrás de la programación para niños y creador del lenguaje Logo, y considerado como el pionero del «pensamiento computacional», introdujo la idea de que la programación puede proporcionar a los niños una manera de pensar sobre su propio pensamiento y aprender sobre su propio aprendizaje. En el contexto de las metodologías de aprendizaje activo, la programación se convierte en una herramienta para planificar, conducir y evaluar problemas y soluciones.

Como se mencionó anteriormente, la investigación académica sobre el PC aumentó en la última década. Muchos autores adoptaron el aprendizaje basado en proyectos, el aprendizaje basado en problemas, el aprendizaje cooperativo y el aprendizaje basado en juegos en PC. Además, todas estas metodologías se basan en la teoría AE. Por lo tanto, las siguientes secciones discutirán primero las diferentes metodologías y enfoques y su conexión e implementación en PC.

### **3.3.2.1 Aprendizaje basado en proyectos en las actividades de PC**

El aprendizaje basado en proyectos (ABP) es una metodología de aula dinámica y centrada en el estudiante. ABP está orientado a resolver temas específicos o proyectos tomados de la vida real, de modo que, al igual que con otras metodologías activas, el estudiante adquiera la capacidad de construir su propio proceso de aprendizaje.

Los estudiantes participan activamente en proyectos del mundo real. Esto se hace a estas cinco características clave:

1. Los maestros proporcionan a sus estudiantes una pregunta de manejo. Esta pregunta debe iniciar el proceso para resolver el problema.
2. Los estudiantes comienzan a explorar esta pregunta de conducción participando en una consulta auténtica y localizada.
3. Estudiantes, profesores y miembros de la comunidad colaboran en actividades para encontrar una solución al problema.

4. Los estudiantes estarán equipados con tecnologías de aprendizaje para ayudarlos a lograr resultados y participar en actividades.
5. Los estudiantes crean un conjunto de productos tangibles que abordan el problema (Krajcik, 2006).

El enfoque se basa en la suposición de que los estudiantes obtienen una comprensión profunda de las materias cuando interactúan con el mundo y experimentan los temas. El enfoque contrarresta la forma tradicional de enseñar donde la información es presentada y transmitida por los profesores. Los defensores de este enfoque argumentan que solo el aprendizaje superficial puede ocurrir en estas formas tradicionales de enseñanza (Krajcik, 2006).

Una clase que hace uso de ABP debe incluir los siguientes pasos:

1. Generar y estimular: Los maestros deben discutir temas con sus estudiantes y activar su curiosidad.
2. Definir y refinar: A partir de este punto, el profesor debe definir una pregunta de conducción que se basa en los intereses y el entorno de sus estudiantes. Estas son preguntas que no se pueden responder con un sí o un no, en general, deben formularse de manera que inicien una investigación, curiosidad y compromiso más profundos.
3. Diseñar y colaborar: En esta etapa, el proyecto está diseñado con la ayuda de los principios SMART (específico, medible, alcanzable, relevante, oportuno). Se asignarán diferentes tareas a los alumnos. No obstante, debería prestarse especial atención a la colaboración.
4. Comparar y compartir: La metodología basada en proyectos hace hincapié en la importancia de la retroalimentación. La retroalimentación debe provenir del profesor y de los compañeros, así que esto se hace creando grupos para comparar y compartir sus ideas.
5. Mejorar y avanzar: Durante la experiencia de aprendizaje, los estudiantes mejoran las habilidades y el conocimiento que ya tienen.

6. Revisar y revisar: Los estudiantes tienen una retrospectiva de lo que han hecho. Se les pide que evalúen su trabajo.
7. Producir y presentar: El producto final debe ser presentado (Harding da Rosa, 2018).

Como recordatorio, el PC es un enfoque de resolución de problemas utilizando algunas de las técnicas informáticas como dividir los grandes problemas en partes más pequeñas, buscando similitudes entre y dentro de los problemas, centrándose solo en información importante y así sucesivamente.

La integración del modelo de aprendizaje basado en proyectos en el pensamiento computacional implica tomar un proyecto como referencia. El estudiante trabajará a través de un proceso de adquisición de conocimientos y así crear nuevos conocimientos como resultado de un proceso de toma de decisiones en el marco de un modelo de prueba/error. Esta es una experiencia eminentemente constructora en la que el estudiante interactúa con el entorno para abordar proyectos del mundo real.

Trabajar en el pensamiento computacional a través del aprendizaje basado en proyectos nos permitirá trabajar en dimensiones como la capacidad de abstracción o la identificación de patrones y variables. Además, nos permitirá expresar ideas y fomentar la creatividad, desarrollando habilidades para prototipar o definir el proyecto.

Como parte esencial del proceso de aprendizaje, este conocimiento debe ser evaluado. El docente se convierte en un facilitador que guía el proceso proponiendo diferentes escenarios que contextualizan el proceso de aprendizaje.

¿Cómo se puede combinar el PC con el aprendizaje basado en proyectos en una clase en línea? Por lo tanto, Shin et al. (2021) propone utilizar los cinco pasos presentados anteriormente y adaptarlos al contexto del PC:

1. Centrarse en los objetivos de aprendizaje. Estos objetivos deben permitir a los estudiantes demostrar el dominio tanto del tema elegido como de la práctica de PC.
2. Comenzando con una pregunta de conducción en tierra en PC. Esta pregunta debe tener una adaptación a las vidas y el entorno de los estudiantes.
3. Explorar la pregunta de conducción participando en trabajos grupales y otras actividades que se intersecan con PC (por ejemplo, hacer preguntas, desarrollar y usar modelos, analizar datos, construir una explicación y diseñar una solución). Es crucial que los estudiantes sean capaces de transferirlo a su vida cotidiana.
4. Creación de un conjunto de productos de PC tangibles.
5. Proporcionar a los estudiantes tecnologías de aprendizaje de PC.

Para implementar un enfoque basado en proyectos en una clase en línea junto con PC, se pueden utilizar herramientas de gestión de proyectos para el liderazgo de equipo. Así, el profesor puede monitorear el proceso de cada grupo. Estas herramientas pueden ser Project Pals, Headrush o Student corner.

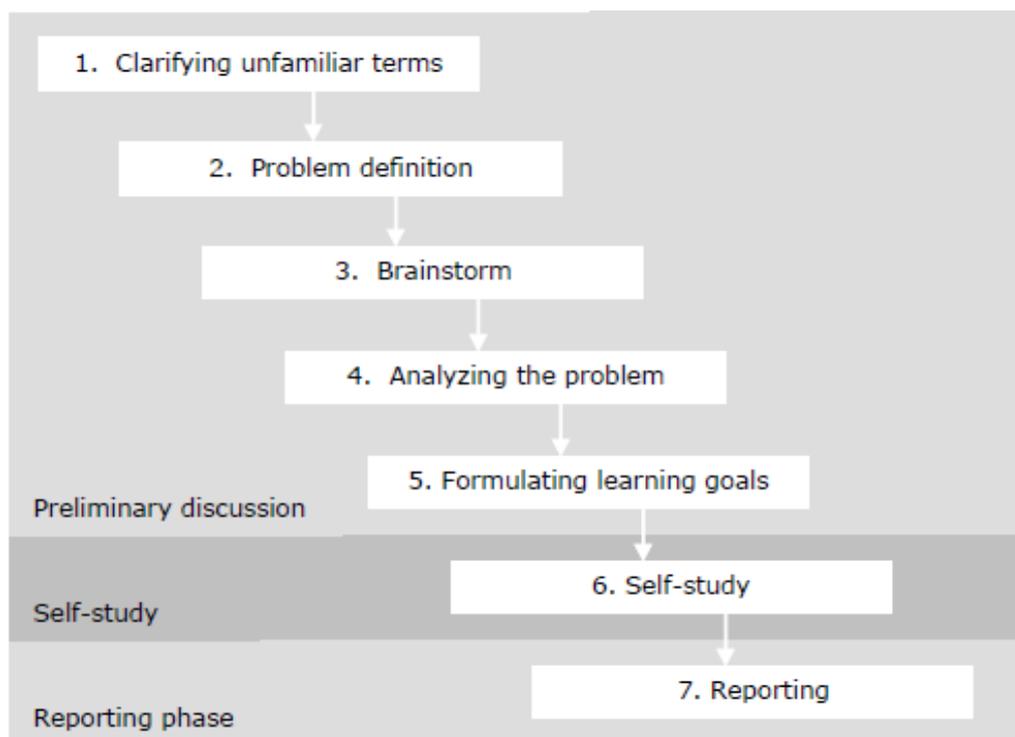
Con todo, mantenga los pasos del enfoque de aprendizaje basado en proyectos e intente diseñar las tareas y la Pregunta Guiadora de una manera que se puedan utilizar los principios de PC. Esto significa que en el segundo paso de la exploración de la cuestión, los niños deben seguir los principios del PC: Rompiendo grandes problemas en pequeños, encontrando igualdades y así sucesivamente.

### **3.3.2.2 Aprendizaje basado en problemas en las actividades de PC**

El aprendizaje basado en problemas (ABP) es una variante del aprendizaje basado en proyectos. Se basa en los mismos principios y características clave que los mencionados anteriormente. Sin embargo, hay algunas diferencias. Con el fin de iniciar una clase de ABP, los estudiantes se enfrentan a

«desencadenantes» de un caso problemático. Sin embargo, es el maestro el que presenta el problema, pero él o ella cambia de un presentador de información a un facilitador de un proceso de resolución de problemas. Esto significa que el aprendizaje basado en problemas tiene como objetivo que los estudiantes se conviertan en estudiantes autodirigidos, pero los maestros todavía los guían monitoreando la discusión e interviniendo cuando sea apropiado (Allen et al., 2011). Por lo tanto, el aprendizaje basado en problemas no es un enfoque que se centra en la resolución de problemas per se, sino en el aumento del conocimiento. La principal diferencia es que en un enfoque de aprendizaje basado en problemas, la educación no tiene lugar en las aulas sino en pequeños grupos de estudiantes donde los estudiantes discuten problemas utilizando el enfoque de siete pasos para encontrar una solución (Wood 2003). El aprendizaje basado en problemas tiene su origen en la educación médica. Aquí, el enfoque se inventó como un proceso tutorial porque el patrón intensivo de conferencias de ciencias básicas seguido de un agotador programa de enseñanza clínica se estaba volviendo ineficaz (Savery, 2006). Por lo tanto, el aprendizaje basado en problemas fue originalmente un enfoque para los estudiantes de medicina en las universidades y se basa en un programa de tutoría donde el maestro guía a los estudiantes a través de la conferencia y donde los estudiantes y el profesor obtienen un papel diferente que en un sistema tradicional.

El aprendizaje basado en problemas se basa en un enfoque de siete pasos:



(Camp et al., 2014)

De acuerdo con estos siete pasos, el profesor prepara un problema auténtico, mal estructurado y del mundo real y establece los objetivos de aprendizaje. Luego, los estudiantes trabajan en pequeños grupos. Tienen la responsabilidad de su propia experiencia de aprendizaje, pero son guiados por el tutor. Existen diferentes enfoques de aprendizaje basados en problemas, pero la mayoría de ellos definen diferentes roles que los estudiantes tienen que obtener en las sesiones con el profesor. Los estudiantes pueden obtener roles como líder del proyecto, grabador (toma notas de cada reunión), miembro del equipo. El primer paso es aclarar términos desconocidos. El objetivo es que todos estén en la misma página. Entonces el problema tiene que ser definido en el segundo paso. El grupo tiene que ponerse de acuerdo sobre el fenómeno que quieren explicar. En la siguiente parte, el grupo pasa a una sesión de lluvia de ideas donde se activa y se determina el conocimiento preexistente y luego analizarán el problema en el siguiente paso. Aquí, los estudiantes tienen que discutir explicaciones e hipótesis. El objetivo de todos es que el grupo de estudiantes formule metas

de aprendizaje al final de la discusión preliminar. A partir de este punto, entran en la fase de auto-estudio. Los miembros del grupo buscan individualmente literatura relevante que pueda responder a las preguntas en los objetivos de aprendizaje y preparar informes para la próxima reunión tutorial. Al final de cada sesión, el tutor y los demás miembros del grupo tienen que dar comentarios sobre los logros (Camp et al., 2014).

Las ventajas del enfoque de aprendizaje basado en problemas son que promueve el autoaprendizaje y es altamente atractivo. Además, ayuda a los estudiantes a desarrollar habilidades transferibles y mejorar las habilidades y habilidades de trabajo en equipo, comunicación e investigación.

Usar ABP para el desarrollo del pensamiento computacional significa comprender el problema, facilitar soluciones e impulsar la toma de decisiones, utilizando una secuencia de fases que se traduciría en pensamiento algorítmico. Las fases principales del proceso serían las siguientes:

- Dividiendo el problema en partes
- Extraer información clave
- Desarrollar modelos descriptivos para entender el problema y buscar una solución
- Crear y probar soluciones automatizadas

**La automaticidad** permitirá que todos los conocimientos se centren en encontrar la solución al problema.

**La activación** nos permitirá identificar los principales elementos/descomponer el problema, analizar sus interacciones y aislar patrones de comportamiento del problema.

**La causalidad** facilita el establecimiento de un sentido de la relación entre el conocimiento que estamos adquiriendo y nuestro conocimiento previo. A partir de ahí es el momento de hacer conexiones entre lo que sabemos y lo que aprendemos.

La **Generación de Conexiones**. Las metodologías activas permiten al estudiante, que está en el centro del proceso de aprendizaje, generar las conexiones que surgen del análisis del problema. El estudiante está habilitado para conectar el conocimiento adquirido con el conocimiento existente que él o ella reunió de antemano. Estamos codificando el aprendizaje y el docente, que, en un contexto de metodologías activas, asume el papel de facilitador, apoya el proceso al permitir que se descomponga en procesos y componentes más pequeños.

**La codificación** se entiende en este contexto como una subtarea de programación que implica la tarea de comprender un problema.

El maestro es el facilitador del proceso, él o ella descompone el problema a través del cuestionamiento

Los estudiantes crean conexiones

El alumno realiza el proceso de análisis mediante la generación de un proceso de codificación para resolver el problema a través del análisis.

**El descubrimiento.** El profesor guía a los estudiantes en el proceso de exploración del problema que les permite identificar relaciones y patrones, descubriendo las reglas y principios subyacentes al problema, permitiendo el descubrimiento de patrones.

**Metacognición.** El objetivo es reflexionar sobre el proceso de aprendizaje. Estudiar el proceso de resolución de problemas nos permitirá compartir el proceso de aprendizaje, generar códigos y detectar posibles errores, analizando cómo hemos desarrollado el proceso (Mayer & Wittrock 1996, 2006).

Tanto en el Aprendizaje Basado en Problemas como en el Aprendizaje Basado en Proyectos, son los estudiantes quienes, normalmente organizados en grupos, establecen su orden de tareas, hacen propuestas de horarios, buscan

información y la aplican en su producto final, difundiendo las posibles soluciones. Todo esto hace uso de la creatividad, el pensamiento crítico y la colaboración.

### **3.3.2.3 Aprendizaje cooperativo en las actividades de PC**

Como se mencionó anteriormente, el aprendizaje cooperativo es un enfoque centrado en el estudiante, método de instrucción facilitado por el instructor en el que los estudiantes trabajan en grupos con la orientación de un profesor. También se centra en la responsabilidad de los estudiantes de su propio aprendizaje. Además, el aprendizaje cooperativo tiene su origen en el constructivismo social que enfatiza que los roles de la cultura y la sociedad, el lenguaje y la interacción son importantes para la forma en que los individuos aprenden (Vygotsky citado por Li & Lam, 2013). El social-constructivismo afirma que el conocimiento es cultural, y el desarrollo de cada individuo es un proceso cultural. Las habilidades que el individuo gana son el resultado de las interacciones sociales con los demás (Li & Lam, 2013).

Por lo tanto, el aprendizaje cooperativo se centra en la disposición de los estudiantes en grupos. Al agrupar a los estudiantes, el profesor tiene que enfocarse para ofrecer algunos elementos estructurales que ayuden a los estudiantes a ver los beneficios de su trabajo en grupo (Mourtos, 1994):

**Formación de equipos:** Al agrupar a los estudiantes en unidades de trabajo, hay dos factores que deben tenerse en cuenta: a. el tamaño de los equipos, y b. los miembros de los equipos. El tamaño del equipo es un factor crítico para la colaboración del grupo. Por un lado, más estudiantes en el grupo significa más información y conocimiento. Pero por otro lado, los grupos más grandes requieren muchas habilidades sociales, y puede causar problemas en la interacción. Por lo tanto, es crucial que el maestro elija el tamaño del grupo sabiamente, pero teniendo en cuenta las características de los miembros del grupo. Los profesores deben conocer bien a sus alumnos para agrupar los

equipos con una mezcla de habilidades que los diferentes estudiantes tienen para ofrecer (Mourtos 1994).

**Interdependencia positiva:** «Para que CL funcione, los estudiantes deben entender que nadan juntos o se hunden juntos» (Mourtos, 1994). La interdependencia positiva significa que los estudiantes tienen la responsabilidad de su propio aprendizaje, pero también del desarrollo y el aprendizaje de sus compañeros de grupo. Estas dos declaraciones tienen que ser claras para los estudiantes.

**Responsabilidad individual:** Sin embargo, el enfoque favorece y promueve las obras grupales que los estudiantes deben tener responsabilidad por sí mismos. Para evitar que cualquier estudiante se convierta en un «ciclista gratuito» y evite participar, los maestros deben hacer lo siguiente:

- Si los estudiantes todavía no están alfabetizados en habilidades cooperativas, mantenga el grupo pequeño
- Realizar pruebas individuales
- Dar tareas individuales a los miembros del grupo. Por lo tanto, son responsables de esta parte, y tienen que enseñar sobre esta parte del proyecto. Este enfoque es el llamado enfoque «jigsaw».
- Los estudiantes deben tener un cuaderno personal (Mourtos, 1994).

Así como en las otras metodologías, el papel de los profesores en el Aprendizaje Cooperativo cambia de profesor a facilitador. El profesor obtiene la gestión de la clase y el proceso de aprendizaje, pero los estudiantes son responsables de los resultados del aprendizaje. Son las fuentes de ideas y soluciones.

¿Cómo combinar el aprendizaje cooperativo con PC? En la implementación de un proyecto de aprendizaje PC de acuerdo con los principios del aprendizaje colaborativo, profesores y estudiantes trabajan juntos trabajando en las dimensiones del proceso, encontrando soluciones y comprendiendo el proceso. El objetivo es:

- Automatizar soluciones
- Recopilar información/datos
- Establecer patrones entre los datos
- Uso del pensamiento paralelo para dar sentido a la información recopilada
- Modelado a través de datos para permitir la experimentación a través del modelado de prueba/error
- Evaluar el proceso como grupo en términos de eficacia y eficiencia

Esto también se puede hacer en línea a través de diferentes sistemas de gestión del aprendizaje y plataformas que ofrecen herramientas de videoconferencia.

### **3.3.2.4 Aprendizaje basado en juegos en las actividades de PC**

Una vez más, los elementos esenciales del aprendizaje basado en juegos se explicarán antes de explicar cómo se pueden vincular el aprendizaje basado en juegos y la PC.

El aprendizaje basado en juegos es un método educativo que utiliza ciertos principios del juego y los transfiere a contextos no relacionados con el juego y configuraciones de la vida real con el objetivo de involucrar al estudiante (Trybus, 2015). El término fue acuñado en el milenio a través del trabajo de autores como James Paul Gee (2007), Diana Oblinger (2006), Richard Van Eck (2006), Steven Johnson (2006) y Marc Prensky (2007). Pero mirando hacia atrás en la historia, los juegos siempre han sido parte de la enseñanza. El ajedrez se utilizaba para enseñar pensamiento estratégico. El aprendizaje basado en juegos es una de las metodologías con mayor impacto en el contexto actual de las metodologías de aprendizaje activo.

El concepto central de la metodología basada en el juego es la enseñanza a través de una forma diferente de la interpretación del fracaso. Los estudiantes trabajan hacia una meta, pero están aprendiendo y estudiando

mientras experimentan los temas a través de juegos o elementos del juego. Tienen que elegir sus acciones y pueden repetirlos tan a menudo como quieran. La metodología basada en el juego es sobre la repetición, el fracaso y los logros de los objetivos (Cahill, n.d).

Las ventajas de esta metodología son que hace que el aprendizaje sea más divertido, por lo tanto aumenta el compromiso de los estudiantes. Además, ofrece a los estudiantes la libertad de fracasar y experimentar. Los errores y el fracaso son parte del proceso de aprendizaje y no se interpretan como algo malo, sino mucho más como algo necesario y como una parte elemental del aprendizaje.

Como hay una gran variedad de juegos, también hay muchas maneras y posibilidades diferentes de cómo los maestros pueden usar los juegos en su aula: Juegos de mesa, Juegos de cartas, Videojuegos, Simulaciones, Juegos de rol y así sucesivamente. Además, la metodología es especialmente fructífera para las clases en línea debido a la alta oferta en Internet. Hay muchas plataformas diferentes que ofrecen elementos basados en juegos. Un ejemplo es Kahoot. Aquí los maestros pueden preparar cuestionarios donde los estudiantes pueden participar con su dispositivo, y obtienen puntos para cada respuesta correcta.

Obviamente, la efectividad de la metodología basada en el juego en el aula depende del diseño de los juegos para lograr diferentes objetivos pedagógicos, así como de las diferentes teorías de aprendizaje incluidas en cada uno de ellos.

¿Cómo combinar PC y aprendizaje basado en juegos? Para responder a esta pregunta, examinaremos una de las posibilidades antes mencionadas para usar juegos en el aula: La simulación.

Los juegos de simulación implican un proceso de aprendizaje inductivo en el que el estudiante investiga activamente para adquirir los objetivos de aprendizaje.

A partir de un marco o escenario específico, el docente introduce elementos con el objetivo de hacer atractivo el proceso y poder aplicar el pensamiento deductivo a través de la resolución de pruebas, que es clave, por ejemplo, en la enseñanza de las asignaturas STEM.

Una vez más, el objetivo es establecer patrones y buscar soluciones utilizando la simulación que el juego permite. La simulación permitirá al profesor representar los datos y procesos del mundo real, con los que quiere que el alumno trabaje, utilizando diferentes modelos para experimentar. La metodología de aprendizaje basado en juegos proporcionará al profesor las posibilidades de trabajar con diferentes escenarios resultantes de la combinación de diferentes parámetros, lo que permitirá combinar la diversidad de los estudiantes y los diferentes entornos de aprendizaje.

De esta manera, el docente promueve el aprendizaje por descubrimiento en un contexto de «Learning by Doing» que promueve la motivación del estudiante, permitiendo la construcción y deconstrucción de situaciones.

La metodología basada en juegos en los contextos PC se desarrolla en torno a estas dimensiones principales:

- Simulaciones
- Experimentación
- Investigación
- Modelado
- Predicciones

### 3.3.3 Los desafíos de la educación en PC y cómo superarlos

#### **Desafíos**

Según muchos expertos, la demanda de educación en línea de los estudiantes aumentó durante el período de veinte años entre 2002 y 2022, pero la aceptación de los maestros de la educación en línea siguió siendo baja, con un 32 % (Allen y Seaman, 2013, 2013). Lloyd, McCoy y Byrne (n.d.) realizaron un

estudio de encuesta sobre las barreras a la educación en línea y encontraron preocupaciones sobre la pedagogía y el apoyo de los maestros, así como la calidad relativa de las ofertas de cursos presenciales vs. en línea (Smidt et al. 2014). Los principales desafíos emergentes se ven a continuación:

**Q: *¿Puede la entrega en línea ofrecer la misma calidad educativa que el aprendizaje presencial?***

A: No podemos apoyar totalmente que los métodos tradicionales de aprendizaje en línea ofrezcan los mismos resultados educativos que el aprendizaje presencial. Pero con CT, la situación podría ser diferente. PC ofrece a los profesores no solo la oportunidad de explorar nuevos procesos de resolución de problemas, sino que también ayuda a desencadenar la creatividad en el diseño de cursos, investigación y políticas dentro de sus áreas de especialización. Una forma de ofrecer a los maestros la oportunidad de aprender PC sería proporcionar la oportunidad de emprender el desarrollo de cursos en línea, utilizando PC como modelo. Es por eso que creemos que en el caso PC, el diseño de un curso en línea ofrece mejores resultados que el aprendizaje presencial.

**Q: *¿La política escolar apoya tanto a los profesores como a los estudiantes en la entrega en línea?***

A: Las autoridades escolares promueven oportunidades en el uso de herramientas tecnológicas para guiar y apoyar la pedagogía y promover el aprendizaje colaborativo en el diseño de cursos en línea. Somos testigos de lo anterior durante el período de pandemia, donde las autoridades escolares, profesores y estudiantes sincronizaron adecuadamente la dirección de proporcionar material de aprendizaje electrónico.



Un ejemplo de problema común es la implementación de tecnología y la capacitación. La capacitación de muchos maestros a la vez puede comprimir el tiempo necesario, pero explotar el costo de la implementación de tecnología en apoyo de los cursos en línea. Categorizar estos y otros problemas similares puede ofrecer claridad y permitir que surja una estrategia eficaz. En el ejemplo de un problema de tecnología y capacitación, a nivel institucional, puede ser posible analizar el problema en categorías tales como los problemas de adopción de decisiones y los problemas de política, mientras que a nivel escolar la aplicación y la capacitación de la tecnología pueden surgir como una cuestión de desarrollo estratégico. Una vez que estos y otros problemas similares son explorados y definidos, la aplicación de estrategias de PC puede permitir un mayor refinamiento de las posibles soluciones.

### **Aplicación de soluciones de pensamiento computacional**

Wing (2006) describió cómo y por qué el Pensamiento Computacional (PC) es un conjunto importante de habilidades para la resolución de problemas. Según Wing, la PC es una forma de conceptualizar los problemas que deben resolver los seres humanos a través de la integración de métodos fundamentales derivados de la informática. En el caso de la gestión del

cambio relacionada con la adopción del aprendizaje en línea, los procesos para superar las barreras pueden derivarse rompiendo problemas específicos e identificando patrones tanto en los problemas como en cualquier solución conocida. La abstracción de ideas que forman principios puede guiar soluciones y crear algoritmos o soluciones paso a paso que ofrezcan claridad lógica. Al pensar de manera diferente sobre cómo los problemas se estructuran y las soluciones se estructuran, el aprendizaje se basa en la teoría, pero se aplica a un proceso relevante y útil.

**Descomposición y recursiva:** Al dividir los problemas en componentes más pequeños, es posible centrar la atención en el tipo de problema y sus problemas de componentes. El uso de la descomposición para desglosar las preocupaciones interpersonales citadas por los profesores en relación con el aprendizaje en línea puede ofrecer claridad en la secuenciación y considerar cada una como parte del todo. Lloyd et al. (n.d.) determinó que los maestros consideran que una pérdida potencial de interacción interpersonal con los estudiantes es una barrera importante para la adopción del aprendizaje en línea. Descompusieron este resultado en cinco categorías específicas de preocupación relacionadas con cambios específicos de interacción social a través de un análisis factorial ponderado.

**Patrones:** La intuición humana impulsa el reconocimiento de patrones. Los patrones comunes comienzan a surgir en nuestra conciencia a medida que aprendemos cualquier nueva habilidad. Moldoveanu, (2009) discute cómo reconocer patrones en problemas puede promover predicciones y soluciones estratégicas. Por ejemplo, los profesores que aprenden a desarrollar cursos en línea no necesitan aprender a codificar cada curso en HTML. Más bien, deberían ser capaces de reconocer los patrones en las diversas estructuras que conforman un sistema de gestión de aprendizaje en línea (LMS) como Canvas, D2L, Moodle o Blackboard. Si bien cada LMS tiene sus propios detalles propietarios únicos, cualquier contenido de curso pedagógicamente sólido se puede modelar en el sistema. Los estudiantes también responden a

patrones repetitivos en la organización del curso dentro de un programa. El reconocimiento de patrones humanos es el resultado de la abstracción y el pensamiento recursivo (pensar en pensar). Pensamos computacionalmente en varios casos cada día sin una clara conciencia de hacerlo.

**La abstracción:** Czerkowski y Lyman (2016) describen la abstracción como «una herramienta que permite la creación de grandes y complejos sistemas de información mediante la definición y generalización de componentes más simples». Esto puede ser útil cuando uno se encuentra con los desafíos descritos por (Silber 2007) en el diseño de entornos y métodos educativos cuando el diseño se guía por problemas «*mal estructurados*» o mal definidos al principio. La abstracción nos permite examinar la estructura y complejidad de los problemas antes de centrarnos en los detalles. «El proceso de pensamiento más importante y de alto nivel en el Pensamiento Computacional es el proceso de abstracción. La abstracción se utiliza en la definición de patrones, generalizando a partir de instancias y parametrización” (Wing 2011). Un proceso común en el desarrollo de cursos académicos es el uso de mapas curriculares. Komenda et al. (2015) discutir el uso del mapeo curricular a través de representaciones espaciales del currículo para que las interconexiones puedan visualizarse gráficamente utilizando análisis de aprendizaje, algoritmos y modelos para comprender completamente los resultados del aprendizaje. Mientras que el currículo de educación secundaria se ve en el nivel de escuela, programa y curso, los maestros pueden participar con el plan de estudios en cada nivel de abstracción, viendo un mapa o red de interconectividad. Voskoglou y Buckley (2012) describen la abstracción como una forma de mapear de una representación compleja a una más simple.

Lu y Fletcher (n.d.) afirman que los problemas pueden ser entendidos y resueltos de manera más efectiva mediante el fomento de múltiples niveles de abstracción. Por ejemplo, si las preocupaciones de política se ven desde las perspectivas de arriba hacia abajo y de abajo hacia arriba, entonces las

preocupaciones administrativas y estudiantiles de la escuela se pueden abordar. Centrarse en un nivel específico de abstracción dentro de un problema puede dar claridad y soluciones más eficientes.

**Paralelismos:** En informática, el procesamiento paralelo se utiliza para llevar a cabo muchas tareas informáticas sincrónicamente; el pensamiento paralelo se presta para definir aún más los problemas y darles sentido cognitivamente. Los proyectos de diseño instruccional a menudo utilizan enfoques secuenciales y paralelos. Por ejemplo, Alsofyani y Baharuddin (2012) describieron sus experiencias guiando a los profesores a través del cambio utilizando el modelo de Conocimiento Tecnológico, Pedagógico y de Contenido (TPACK) para construir competencias en la educación en línea. Descubrieron que aunque estudios previos han explorado un modelo híbrido de aprendizaje en línea en la enseñanza con tecnología, los participantes calificaron favorablemente una experiencia de desarrollo de profesores totalmente en línea. Del mismo modo, (Rienties et al. 2013) examinó el impacto de la formación colaborativa de profesores en diferentes instituciones y disciplinas fuera de las suyas en un curso paralelo de formación docente en tecnología.

**Diseño de algoritmos:** Los algoritmos básicos son procesos paso a paso para resolver un desafío específico. Usamos algoritmos cada día en nuestro pensamiento en simples intercambios humanos como la conducción. Si la luz es roja, debo detenerme, si es amarilla, tengo opciones. Con respecto a la formación de profesores algoritmos para resolver problemas, Moldoveanu, (2009) discute cómo los algoritmos pueden proporcionar pasos lógicos, y también varios modelos diferentes en la elección de una solución viable. Por ejemplo, con respecto a la formación técnica, no todos los participantes aprenden tecnología al mismo ritmo. Ofrecer un enfoque paso a paso en un foro de aprendizaje en línea puede promover la conciencia de la facultad de cómo los estudiantes pueden percibir el aprendizaje en un formato en línea y

construir comodidad con su propia experiencia en un sistema de gestión del aprendizaje.

## Referencias

Ahmadi, Mohammad Reza and Shahid Beheshti University, Tehran, Iran. 2017. "The Impact of Motivation on Reading Comprehension." *International Journal of Research in English Education* 2(1):1-7. doi:

10.18869/acadpub.ijree.2.1.1.

Ahmadi, Navid, and Mehdi Jazayeri. 2014. "Analyzing the Learning Process in Online Educational Game Design: A Case Study." Pp. 84-93 in 2014 23rd *Australian Software Engineering Conference*. Milsons Point, NSW, Australia: IEEE.

Aho, A. V. 2012. "Computation and Computational Thinking." *The Computer Journal* 55(7):832-35. doi: 10.1093/comjnl/bxs074.

Allen, E., & Seaman. (2013). Changing course: Ten years of tracking online education in the United States. CA: Babson Survey Research Group & Quahog Research Group, LLC. <http://ezproxy.library.arizona.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=E D 541571&site=ehost-live>

ALLEN, D. E., DONHAM, R. S., & BERNHARDT, S. A. (2011). Problem-based learning. *New directions for teaching and learning*, 2011(128), 21-29.

Alsofyani, Mohammed Modeef, and Aris Baharuddin. 2012. "A Preliminary Evaluation of Short Blended Online Training Workshop for TPACK Development Using Technology Acceptance Model." *Turkish Online Journal of Educational Technology*.

ANGELI, C., & GIANNAKOS, M. (2020). Computational thinking education: Issues and challenges. *Computers in Human Behavior*, 105, 106185.

Barr, Valerie, and Chris Stephenson. 2011. "Bringing Computational Thinking to K-12: What Is Involved and What Is the Role of the Computer Science Education Community?" *ACM Inroads* 2(1):48–54. doi: 10.1145/1929887.1929905.

Benkler, Y. (n.d.). *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press.  
10.12987/9780300127232

BERGMANN, J. & SAMS, A. (2012). Before you flip, consider this. *Phi Delta Kappa International*. Vol. 94 (2), p. 25.

BERGMANN, J. & SAMS, A. (2012). Flip Your Classroom: Reach Every Student in Every Class Every Day.

Brennan, Karen, and Mitchel Resnick. 2012. "New Frameworks for Studying and Assessing the Development of Computational Thinking." 25.

CAHILL, G. (N.D.). Why Game-based Learning?, online URL: <https://thelearningcounsel.com/article/why-game-based-learning>.

CAMP, G. ET AL (2014): PBL: step by step. A guide for students and tutors, online URL: [https://teach.its.uiowa.edu/sites/teach.its.uiowa.edu/files/docs/docs/Steps\\_of\\_PBL\\_ed.pdf](https://teach.its.uiowa.edu/sites/teach.its.uiowa.edu/files/docs/docs/Steps_of_PBL_ed.pdf).

CARVER, R. (1996). Theory for practice: A framework for thinking about experiential education. *Journal of Experiential Education*, 19(1), 8-13.

Cheliotis, G., Hu, N., & Yew, J. (2014) . *The Antecedents of Remix*. In *ACM 17th Conference on Computer Supported Cooperative Work*, 38.  
<http://dx.doi.org/10.1145/2531602.2531730>

Chi, Michelene T. H., and Muhsin Menekse. 2015. "Dialogue Patterns in Peer Collaboration That Promote Learning." Pp. 263–74 in *Socializing Intelligence*

*Through Academic Talk and Dialogue*, edited by L. B. Resnick, C. S. C. Asterhan, and S. N. Clarke. American Educational Research Association.

Committee for the Workshops on Computational Thinking, National Research Council (U. S. ), Computer Science and Telecommunications Board, National Research Council (U.S.), and *Air Force Science and Technology Board*. 2010. Report of a Workshop on the Scope and Nature of Computational Thinking. Washington, D.C.: National Academies Press.

Curzon, Paul, and Mark Dorling. 2014.

"DevelopingComputationalThinkingInTheClassroomaFramework.Pdf."

Czerkawski, Betul C., and Eugene W. Lyman. 2016. "An Instructional Design Framework for Fostering Student Engagement in Online Learning Environments." *TechTrends* 60(6):532–39. doi: 10.1007/s11528-016-0110-z.

Dhatsuwan, Apinya, and Monamorn Precharattana. 2016. "BLOCKYLAND: A Cellular Automata-Based Game to Enhance Logical Thinking." *Simulation & Gaming* 47(4):445–64. doi: 10.1177/1046878116643468.

European Commission. Joint Research Centre. 2016. Developing Computational Thinking in Compulsory Education: *Implications for Policy and Practice*. LU: Publications Office.

EFSTRATIA, D. (2014). Experiential education through project based learning. *Procedia-social and behavioral sciences*, 152, 1256-1260.

FLIPPED LEARNING NETWORK (FLN). (2014) The Four Pillars of F-L-I-P™.

GENTRY, J. W. (1990). What is experiential learning. *Guide to business gaming and experiential learning*, 9, 20.

GAO, X., & HEW, K. F. (2022). Toward a 5E-Based Flipped Classroom Model for Teaching Computational Thinking in Elementary School: Effects on Student Computational Thinking and Problem-Solving Performance. *Journal of Educational Computing Research*, 60(2), 512-543.

HARDING DA ROSA, J. (2018). Promoting Project-Based Learning, online URL: <https://oupeltglobalblog.com/tag/jane-maria-harding-da-rosa/>.

HSU T.-C., CHANG S.-C. & HUNG Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature, *Computers & Education*.

Ioannidou, Andri, Vicki Bennett, Ashok Basawapatna, Alexander Repenning, and Kyu Han Koh. 2011. *"Computational Thinking Patterns."* American Educational Research Association.

Jenkins, H., Clinton, K., Purushotma, R., & Robinson, A. (2006). *Confronting the Challenges of Participatory Culture: Media Education for the 21st Century*. MacArthur Foundation. [http://digitallearning.macfound.org/site/apps/nlnet/content2.aspx?c=enJLKQNIFiG&b=2108773&content\\_id=](http://digitallearning.macfound.org/site/apps/nlnet/content2.aspx?c=enJLKQNIFiG&b=2108773&content_id=)

Johnson, Terri, Mary Ann Wisniewski, Greg Kuhlemeyer, and Jamie Krzykowski. 2012. "TECHNOLOGY ADOPTION IN HIGHER EDUCATION: OVERCOMING ANXIETY THROUGH FACULTY BOOTCAMP." *Carroll University 16(Journal of Asynchronous Learning Networks):63-72*.

Karram, O. (2021). The Role of Computer Games in Teaching Object-Oriented Programming in High Schools-Code Combat as a Game Approach. *WSEAS Transactions on Advances in Engineering Education*, 18, 37-46.

Koh, Kyu Han, Ashok Basawapatna, Vicki Bennett, and Alexander Repenning. 2010. "Towards the Automatic Recognition of Computational Thinking for Adaptive Visual Language Learning." Pp. 59-66 in 2010 IEEE *Symposium on Visual Languages and Human-Centric Computing*. Leganes, Madrid, Spain: IEEE.

KOLB, D. A. (1984). *Experiential learning: experience as the source of learning and development*. Englewood Cliffs, NJ: Prentice Hall.

Komenda, Martin, Martin Vít, Christos Vaitsis, Daniel Schwarz, Andrea Pokorná, Nabil Zary, and Ladislav Dušek. 2015. "Curriculum Mapping with Academic Analytics in Medical and Healthcare Education" edited by K. E. Vrana. PLOS ONE 10(12):e0143748. doi: 10.1371/journal.pone.0143748.

KRAJCIK, J. S., & BLUMENFELD, P. C. (2006). *Project-based learning* (pp. 317-34).

Krugel, Johannes, and Peter Hubwieser. 2017. "Computational Thinking as Springboard for Learning Object-Oriented Programming in an Interactive

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge University Press.

Lessig, L. (2008). *Remix: Making Art and Commerce Thrive in the Hybrid Economy*. Penguin Press.

LI, M. P., & LAM, B. H. (2013). Cooperative learning. *The Hong Kong Institute of Education*, 1-33.

Lloyd, Steven, Tami McCoy, and Michelle Byrne. n.d. "Faculty-Perceived Barriers of Online Education." *MERLOT Journal of Online Learning and Teaching*.

Lloyd, S., McCoy, T., & Byrne, M. (n.d.). Faculty-Perceived Barriers of Online Education. *MERLOT Journal of Online Learning and Teaching*.

Lu, James J., and George H. L. Fletcher. n.d. "Thinking About Computational Thinking." 5.

MASON, S. L., & RICH, P. J. (2019). Preparing elementary school teachers to teach computing, coding, and computational thinking. *Contemporary Issues in Technology and Teacher Education*, 19(4), 790-824

MASSACHUSETTS DEPARTMENT OF ELEMENTARY AND SECONDARY EDUCATION (DOE). (2016). *Massachusetts Curriculum Framework for Science and Technology/Engineering*.

Manovich, L. (2020, April 15). Remix and remixability. Manovich. Retrieved July 15, 2022, from [http://www.manovich.net/DOCS/Remix\\_modular.doc](http://www.manovich.net/DOCS/Remix_modular.doc)

Moldoveanu, M. 2009. "Thinking Strategically about Thinking Strategically: The Computational Structure and Dynamics of Managerial Problem Selection and Formulation." *Strategic Management Journal* 30(7):737–63.

MOOC." Pp. 1709–12 in 2017 IEEE *Global Engineering Education Conference (EDUCON)*. Athens, Greece: IEEE.

MOURTOS, N. J. (1994, November). The nuts and bolts of cooperative learning in engineering. In *Proceedings of 1994 IEEE Frontiers in Education Conference-FIE'94* (pp. 624-627). IEEE.

NATIONAL RESEARCH COUNCIL (NRC) (2012). *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas*. Washington, DC: The National Academies Press.

Online Learning Statistics: *Is Online Education the Future?* (2021). Bay Atlantic University. Retrieved July 17, 2022, from <https://bau.edu/blog/online-learning-statistics/>

Papert, Seymour. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.

Pellas, Nikolaos, and Efstratios Peroutseas. 2016. "Gaming in Second Life via Scratch4SL: Engaging High School Students in Programming Courses." *Journal of Educational Computing Research* 54(1):108–43. doi: 10.1177/0735633115612785.

Rienties, Bart, Natasa Brouwer, Katerina Bohle Carbonell, Danielle Townsend, Anne-Petra Rozendal, Janneke van der Loo, Peter Dekker, and Simon

Lygo-Baker. 2013. "Online Training of TPACK Skills of Higher Education Scholars: A Cross-Institutional Impact Study." *European Journal of Teacher Education* 36(4):480–95. doi: 10.1080/02619768.2013.801073.

Rob-Bot! The Computational Thinking Card Game! — Rob-Bot Teaching Resources - Teaching Resources for Computing. (2022). Retrieved 26 July 2022, from <https://robotresources.com/blog/2019/2/1/computational-thinking-skills>

SAVERY, J. R. (2015). Overview of problem-based learning: Definitions and distinctions. *Essential readings in problem-based learning: Exploring and extending the legacy of Howard S. Barrows*, 9(2), 5-15.

Shute, Valerie J., Chen Sun, and Jodi Asbell-Clarke. 2017. "Demystifying Computational Thinking." *Educational Research Review* 22:142–58. doi: 10.1016/j.edurev.2017.09.003.

SHIN, N., BOWERS, J., KRAJCIK, J., & DAMELIN, D. (2021). Promoting computational thinking through project-based learning. *Disciplinary and Interdisciplinary Science Education Research*, 3(1), 1-15.

Silber, K. 2007. "A Principle-Based Model of Instructional Design: A New Way of Thinking about and Teaching ID." *Educational Technology*, 5–19.

SLAVICH, GEORGE M. & PHILIP G. ZIMBARDO. 2012. "Transformational Teaching: Theoretical Underpinnings, Basic Principles, and Core Methods." *Educational Psychology Review* 24, no. 4: 569–608.

Smidt, Esther, Brian McDyre, Jennifer Bunk, Rui Li, and Tanya Gatenby. 2014. "Faculty Attitudes about Distance Education." *IAFOR Journal of Education* 2(2). doi: 10.22492/ije.2.2.06.

Social Constructivism & Social Media | socialmedia4444. (n.d.). socialmedia4444. Retrieved July 15, 2022, from <https://socialmedia4444.wordpress.com/social-constructivism-social-media/>

Soh, Leen-Kiat, Duane F. Shell, Elizabeth Ingraham, Stephen Ramsay, and Brian Moore. 2015. "Learning through Computational Creativity." *Communications of the ACM* 58(8):33–35. doi: 10.1145/2699391.

TRYBUS, J. (2015). Game-based learning: What it is, why it works, and where it's going. New Media Institute.

Voskoglou, Michael Gr., and Sheryl Buckley. 2012. "Problem Solving and Computational Thinking in a Learning Environment." doi: 10.48550/ARXIV.1212.0750.

Weintrop, David, Elham Beheshti, Michael Horn, Kai Orton, Kemi Jona, Laura Trouille, and Uri Wilensky. 2016. "Defining Computational Thinking for Mathematics and Science Classrooms." *Journal of Science Education and Technology* 25(1):127–47. doi: 10.1007/s10956-015-9581-5.

Wing, Jeannette M. 2006. "Computational Thinking."

Wing, Jeannette M. 2011. "Research Notebook: Computational Thinking--What and Why?" 8.

WOOD, D. F. (2003). Problem based learning. *Bmj*, 326(7384), 328-330.

Wolz, Ursula, Meredith Stone, Sarah M. Pulimood, and Kim Pearson. 2010. "Computational Thinking via Interactive Journalism in Middle School." P. 239 in Proceedings of the 41st ACM technical symposium on Computer science education - SIGCSE '10. Milwaukee, Wisconsin, USA: ACM Press.

Yu, L., & Nickerson, J. V. (2011). Cooks or cobblers?: crowd creativity through combination. In ACM 2011 annual conference on Human factors in computing systems (CHI '11), 1393–1402.

<http://dx.doi.org/10.1145/1978942.1979147>

## Capítulo 4: Formación de Profesores en Pensamiento Computacional: Elementos de innovación e impacto esperado

---

### 4.1. Desarrollo profesional eficaz del profesorado (TPD)

La enseñanza es un compromiso profesional muy especial. Como el papel del docente se considera multidimensional, un maestro contribuye a la difusión del conocimiento y potencia el crecimiento de los estudiantes, ayudándoles a desarrollar el aspecto espiritual, social y psicológico de su personalidad, lo que hace que su presencia en la vida del estudiante sea crucial. Los profesores actúan como modelos a seguir. Muchos estudiantes son influenciados por ellos y a menudo confían en ellos más de lo que confían en su familia. Esto aumenta la responsabilidad del profesor hacia los estudiantes.

Por otro lado, no debemos pasar por alto la dificultad de la enseñanza. Su aspecto más fundamental radica en la creación de una interacción armoniosa y efectiva entre educadores y aprendices entre diferentes personas de diferentes edades (adultos y menores), de diferentes personalidades con diferentes experiencias. Además, la enseñanza exitosa básicamente requiere la cooperación de diferentes maestros, cada uno de los cuales tiene su propio temperamento, puntos de vista y filosofía con respecto a la educación. Sin embargo, principalmente, la dificultad aumenta teniendo en cuenta los diferentes antecedentes cognitivos de cada estudiante, su entorno social específico, las experiencias desagradables que pueden tener en su vida y otras características particulares que determinan su actitud y rendimiento académico (Knight & Wiseman, 2005)

Todos estos factores dejan claro que la enseñanza no puede y no debe tener un carácter estático. El proceso de aprendizaje debe ser continuo e interminable, un procedimiento de por vida. (Knight, 2002)

Vale la pena enfatizar que los modelos de enseñanza centrados en el estudiante han sido muy efectivos hasta ahora. Cambiar el centro de la enseñanza de los profesores a los estudiantes ayudó a los niños a ser más seguros de sí mismos, los animó a descubrir conocimientos y les mostró cómo construir nueva información sobre la base de su experiencia previa. Los estudiantes dejaron de ser estudiantes pasivos y los maestros dejaron de ser los expertos de la clase (Young & Paterson, 2007). Además, el aprendizaje activo brinda a los educadores la oportunidad de aplicar métodos de enseñanza cooperativos y ayudar a todos los niños a participar en el proceso de enseñanza. Con el apoyo del profesor, la experiencia de aprendizaje se hizo agradable y el interés de los estudiantes en la escuela se incrementó.

Como resultado de este cambio de paradigma en nuestros días, los maestros han adquirido nuevos roles. El educador como investigador tiene la flexibilidad de innovar, inspirado en la individualidad de los estudiantes, en sus talentos, sus deseos, así como en sus debilidades. Este es «el desarrollo profesional definitivo», según Dorothy Suskind (Suskind, 2016). En lugar de estar limitados por el plan de estudios, los maestros «forman» el plan de estudios ellos mismos, comenzando su día con la pregunta «¿Qué me enseñarán mis alumnos hoy?» y convertirse en «directores de su propio desarrollo profesional».

La reciente pandemia de Covid-19 estableció una realidad educativa totalmente mediada por computadora en todo el mundo.

Según UNICEF, «desde el estallido de la COVID-19 hasta la fecha, la pandemia tuvo un impacto devastador en la educación de millones de niños en todo el mundo, lo que la convirtió en la mayor crisis mundial para los niños» en los 75 años de historia de esta organización.

Durante esta pandemia, la educación de todos los grados dependía totalmente de los métodos de aprendizaje electrónico. El aula se convirtió en digital, y sus miembros estaban interactuando solo a través de diferentes dispositivos

digitales. Por lo tanto, el aprendizaje mediado por la tecnología (AMT) y la comunicación se difundieron ampliamente como la única e inevitable herramienta de interacción de estudiantes y educadores. Por lo tanto, debemos considerar orientar el desarrollo profesional de los docentes de acuerdo con las necesidades actuales de los estudiantes y las demandas sociales. Sin embargo, aunque el AMT en línea aparece con ventajas, como el ahorro de tiempo, la flexibilidad y la independencia del ritmo de aprendizaje (Postholm, 2007), aún no está claro si los resultados del aprendizaje son deseables y más efectivos para los estudiantes. El aprendizaje a distancia carece de la necesaria interacción cara a cara entre estudiantes y profesores, pero por otro lado, ofrece rutas alternativas de enseñanza a través de herramientas tecnológicas en línea que hacen que la lección sea amigable y fascinante. Por ejemplo, en entornos de aprendizaje síncronos, los estudiantes tienen la capacidad de comunicarse y cooperar en grupos más pequeños que explotan las herramientas digitales ofrecidas (Centro para la Innovación Académica, 2020).

Por lo tanto, la situación confirma claramente la tendencia creciente a las estrategias de aprendizaje electrónico que el TPD debería abordar inevitablemente. Sin embargo, también hay otra cuestión que debe tenerse en cuenta: o nuestra clase es convencional o digital, es obvio que los educadores tienen que enseñar a sus estudiantes cómo aprender por su cuenta. Los estudiantes necesitan adquirir habilidades metacognitivas que aumenten su conciencia, motivación interna y responsabilidad personal en el proceso educativo. Deben ser capaces de conectar sus conocimientos previos con la nueva información adquirida para alcanzar una comprensión profunda. La metacognición influye en sus habilidades de pensamiento crítico (Magno, 2006). Esta es la clave para un proceso de aprendizaje continuo y permanente, que es necesario para todos. Dado el multiculturalismo de las aulas y el nivel cognitivo calificado de los estudiantes, el desarrollo de habilidades metacognitivas es un factor crucial en la adquisición de conocimiento y en la enseñanza efectiva.

¿Cómo pueden los estudiantes obtener esta metacognición a través de entornos de aprendizaje basados en la web? ¿Cómo pueden los educadores apoyar a los estudiantes a desarrollar el aprendizaje autónomo, la autodisciplina y la confianza en sí mismos a través de estos entornos? ¿Cómo pueden los educadores apoyar a los estudiantes para transformar un problema complejo en otros más pequeños? Finalmente, ¿qué estrategias tienen que aplicar los educadores? La respuesta es sugerida por las disciplinas del Pensamiento Computacional.

#### **4.2 Formación de profesores en pensamiento computacional: Desafíos y Oportunidades**

En los últimos años, un número creciente de países ha introducido el Pensamiento Computacional (PC) en sus planes de estudios nacionales como un medio para alinear la educación con las necesidades de nuestras sociedades ricas en conocimiento de la era de la información y los avances en la tecnología moderna (Saidin, Khalid, Martin, Kuppusamy y Munusamy, 2021; Scherer, 2016). Este desarrollo ha sido respaldado por los resultados de muchos estudios de investigación que demuestran que la Ciencia de la Computación y la Programación de Computación pueden mejorar la participación de los estudiantes, la motivación, la confianza, las habilidades de resolución de problemas, las habilidades de comunicación y mejorar su rendimiento general en ciencia, tecnología, ingeniería y matemáticas (STEM) (Lye & Koh, 2014; Mason & Rich, 2019; Pala & Türker, 2021; Scherer, 2016). Según Wing (2006), en esencia, PC es la capacidad de pensar como un informático para formular un problema y expresar su solución de una manera que podría ser procesada por una computadora.

Sin embargo, PC tiene una definición más amplia que se centra en el proceso de abstracción, pero también incluye, entre otros, conceptos como pensar recursivamente, descomposición de problemas, recopilación de datos, representación y análisis, pruebas y verificación, análisis y validación de modelos, pensamiento algorítmico, evaluación, generalización, simulación y

automatización (Barr & Stephenson, 2011, Wing, 2006). La integración de las habilidades de PC en cualquier campo de la educación, además de las Ciencias de la Computación (CC), ha demostrado ser una tarea bastante desafiante y la formación del profesorado en PC parece ser un factor clave para que cualquier intento de este tipo sea fructífero (Denning, 2017; Saidin et al, 2021).

#### 4.2.1 Desafíos para PC en la Educación

La comprensión del PC por parte de los maestros se considera críticamente importante para introducir de manera significativa sus conceptos en el campo de la educación (Denning, 2017; Saidin et al, 2021). Como resultado de su vaga definición, los educadores parecen luchar con la pregunta de «¿Qué es el Pensamiento Computacional?» y a menudo se les ofrecen definiciones vagas o incluso confusas en respuesta (Denning, 2017; Saidin et al, 2021). Selby y Woollard (2014) argumentan que existe la necesidad de una definición sólida y clara de PC con el fin de facilitar la enseñanza de sus conceptos en las aulas reales de una manera creativa y significativa. Una definición bien fundamentada de PC es esencial para los educadores, ya que no solo les permitiría enseñar o demostrar en la práctica los conceptos de PC en sus clases, sino que también les permitiría desarrollar herramientas de evaluación para medir el nivel de habilidades de sus estudiantes y realizar un seguimiento de su progreso (Saidin et al, 2021; Selby & Woollard, 2014).

Como subraya Denning (2017), los educadores no pueden ser efectivos si no están seguros de lo que están enseñando. Parece que la mayoría de los profesores de pre-servicio y en servicio no han asistido a ninguna formación formal en PC y parecen estar inciertos acerca de los conceptos que están involucrados en ella, así como sus dominios aplicables (Saidin et al, 2021). Esta falta de conocimiento ha llevado a muchos educadores a creer falsamente que el PC es una habilidad estrictamente relacionada con los campos de la CC (Ciencia de la Computación) y la tecnología (Saidin et al, 2021). Heintz, Mannila y Farnqvist (2016) argumentan que existe una fuerte necesidad de

desarrollar las competencias digitales de los profesores junto con las habilidades de programación a través de cursos especializados o programas de formación y de alentar a más educadores a incluir aspectos de la CC en sus asignaturas. En este contexto, se considera extremadamente importante que los educadores comprendan plenamente el contenido que están enseñando, la pedagogía relacionada con ella y la tecnología que están utilizando (Mason & Rich, 2019). Con respecto a la introducción del PC en las clases K12, Shailaja y Sridaran (2015) señalan que la mayoría de los planes de estudio nacionales se ocupan principalmente de «por qué» y «qué» enseñar y la parte «cómo» falta en su contenido.

Parece que preparar a los profesores para incorporar PC en sus clases de K12 y aplicar sus conceptos en varias áreas temáticas es un proceso de aprendizaje que requiere mucho tiempo ya que los educadores no solo necesitan ser entrenados en estrategias de integración de PC para sus cursos, sino que también necesitan mucha práctica práctica del mundo real para dominarlo (Denning, 2017† Mason & Rich, 2019† Saidin et al, 2021). Además, convertirse en competente en conceptos de PC, habilidades y entornos de aprendizaje como la programación, la computación o la robótica educativa puede ser realmente difícil y requiere mucho tiempo para los educadores que son menos competentes en el uso de tecnologías digitales o que nunca han estado expuestos al PC antes o incluso carecen de confianza debido a su limitado conocimiento pedagógico en la enseñanza del PC (Saidin et al, 2021† Mason & Rich, 2019).

Aparte de la competencia digital de los profesores y las habilidades de enseñanza de PC, la falta de materiales didácticos relevantes y apropiados para la edad y estrategias de evaluación para las habilidades de PC de los estudiantes parecen ser las principales barreras para introducir eficazmente los conceptos de PC en las clases K12 de manera interdisciplinaria (Barr & Stephenson, 2011. Saidin et al, 2021). Los educadores capacitados en PC

tienden a imaginarlo como una metodología de resolución de problemas que se puede aplicar en varias materias y reconocen los beneficios de introducir el PC en su aula, pero implementar e integrar actividades de enseñanza y aprendizaje de PC en los planes de estudio existentes de su escuela es un gran desafío para ellos, principalmente debido a las restricciones de tiempo y el tiempo extra que necesitan dedicar (Barr & Stephenson, 2011 \ Denning, 2017} Garneli, Giannakos, & Chorianopoulos, 2015 Saidin et al, 2021). En este contexto, Kakavas y Ugolini (2019) sugieren que clasificar los conceptos de PC que los estudiantes pueden aprender en cada grado, establecer objetivos cognitivos claros al proporcionar ejemplos específicos junto con actividades de enseñanza/aprendizaje a los educadores y desarrollar herramientas de evaluación de PC válidas y confiables, podría promover efectivamente la integración de PC en los currículos de K12.

Las actitudes de los maestros hacia STEM y PC también se consideran factores importantes para promover e integrar eficazmente el PC en los currículos de K12 (Saidin et al, 2021' Yadav, Hong, & Stephenson, 2016). Un estudio experimental realizado por Yadav, Mayfield, Zhou, Hambrusch y Korb (2014) examinó el impacto de una clase de PC de una semana en las actitudes de los maestros hacia la introducción del PC en su aula y los investigadores encontraron que los maestros que asistieron no solo tenían una mejor comprensión de los conceptos/habilidades de PC, sino que también eran más positivos y abiertos al incorporar el PC en sus futuras clases en comparación con los maestros del grupo de control».

#### **4.2.2 Beneficios de la formación del profesorado en PC**

Por difícil que sea, la integración exitosa de conceptos/habilidades de PC en los currículos K12 definitivamente vale la pena hacer un esfuerzo, ya que el PC puede proporcionar muchos beneficios tanto a los estudiantes como a los educadores. Los efectos positivos de la introducción de PC en la educación K12 incluyen mejorar el pensamiento crítico y analítico entre los estudiantes, promover la educación STEM, cambiar las actitudes de los maestros y los

estudiantes hacia ella y mejorar aún más la pedagogía y los planes de estudio (Saidin et al, 2021; Mason & Rich, 2019). Parece que una vez que los estudiantes comienzan a pensar computacionalmente son capaces de desempeñarse mejor en las actividades de resolución de problemas en clase y en situaciones de la vida cotidiana, ya que son capaces de comprender y expresar mejor un problema y su solución adecuada mediante la aplicación efectiva de conceptos como la abstracción, la descomposición, el diseño algorítmico, la generalización y la evaluación (Selby & Woollard, 2014; Mason & Rich, 2019).

Además, teniendo en cuenta la amplia definición de PC de Wing (2006), se podrían hacer afirmaciones similares para los maestros de escuela con capacitación en PC. Un estudio de Pala y Türker (2021) muestra que la participación en una clase de programación de Arduino había tenido un impacto positivo en la creatividad, el pensamiento crítico, el pensamiento algorítmico y las habilidades de PC de los maestros preservicio. A medida que las necesidades sociales cambian rápidamente, impulsadas por los avances en la tecnología, el PC se está convirtiendo en un «conjunto de habilidades» esencial que todos, no solo los informáticos, deberían tener (Pala & Türker, 2021; Wing, 2006). En cuanto a las profesiones docentes, los educadores con habilidades de PC pueden ser más creativos en la enseñanza de las generaciones más jóvenes y, como tienen una mayor confianza en sus habilidades de PC, se puede esperar que tengan un mayor compromiso y satisfacción laboral (Saidin et al, 2021). Además, alentar a los educadores a desarrollar o mejorar sus competencias digitales y habilidades de PC puede ser un pequeño paso hacia la creación de un sistema educativo más relevante, adecuado a las necesidades de las sociedades modernas (Saidin et al, 2021; Heintz, Mannila, & Farnqvist, 2016).

#### **4.3 Aprendizaje Mediado Tecnológico (AMT) y Pensamiento Computacional (PC) para el Desarrollo Profesional del Educador**

Como ya se mencionó, PC utiliza posibilidades computacionales avanzadas para comprender y resolver problemas complejos, exhibiendo una forma

específica de pensar y, por lo tanto, se une con AMT a través de la habilidad necesaria de alfabetización tecnológica.

La alfabetización tecnológica puede definirse mediante dos habilidades básicas:

**a. conocimiento** y utilización de las funciones básicas relativas a las computadoras

**b. uso** de la tecnología con el fin de aumentar la productividad y lograr el desarrollo profesional de los profesores, con el fin de apoyar su enseñanza mediante la utilización de la tecnología y mediante el diseño e implementación de actividades y aplicaciones. (Borthwick, Hansen 2017)

Según el Consejo para la Acreditación de Normas de Preparación de Educadores (CAEP), la Competencia Tecnológica define la capacidad de un educador para «modelar y aplicar estándares tecnológicos a medida que diseñan, implementan y evalúan experiencias de aprendizaje para involucrar a los estudiantes y mejorar el aprendizaje, y enriquecer la práctica profesional» (CAEP, 2015, Faloon, 2000).

#### 4.3.1 Programas de desarrollo profesional mediado por la tecnología (programas TTMPD)

La enseñanza y el aprendizaje mediados por la tecnología (Oliver & Herrington, 2003) establece como requisito previo la alfabetización tecnológica y se refiere al entorno digital que puede ser utilizado por los educadores para tener éxito en la máxima interacción entre ellos y los estudiantes y lograr los resultados de aprendizaje dentro del plan de estudios aplicado. (Bower, 2019).

Algunos ejemplos de métodos y técnicas de enseñanza que se pueden practicar en entornos digitales que incorporan AMT son los siguientes:

- aprendizaje entre pares
- trabajo en equipo
- uso de recursos, vídeos y simulaciones basados en la web
- utilización de plataformas electrónicas y bases de datos

- entornos interactivos de laboratorio digital para lecciones de ciencia
- juegos y analogías
- uso de software para realizar cuestionarios y encuestas cruzadas, o cuestionarios de elección múltiple

Todos estos métodos parecen promover las habilidades de comunicación y colaboración y proporcionar a los maestros las herramientas adecuadas para crear lecciones basadas en la investigación que ayudarán a sus estudiantes a desarrollar estrategias, pensar lógicamente y entrenar su pensamiento crítico. Los resultados del aprendizaje deben clasificarse en el desarrollo psicomotor de los estudiantes, la obtención de objetivos cognitivos y, finalmente, el entretenimiento y la satisfacción durante el procedimiento de aprendizaje. De esa manera los estudiantes habrán aprendido... ¡cómo aprender!

Está claro, después de todo, que para lograr grandes resultados de aprendizaje, los maestros deben tener la posibilidad de recibir acceso a programas educativos de Desarrollo Profesional de Tecnología (TPD). Y dado que la investigación en el campo ha mostrado resultados decepcionantes con respecto a los programas educativos convencionales, basados en conferencias de instructores expertos, la necesidad de programas que impliquen la colaboración social y proporcionar a los maestros acceso a redes de apoyo entre pares ha ganado el juego (Adsit N.J. AACTE, 2004).

Sin embargo, los programas de TPD para docentes deben incluir capacitación sobre la incorporación de las TIC para lo siguiente:

- participación en redes digitales a través de comunidades de aprendizaje que utilizan el modelo de «aprendizaje situado» de Vigotsky.
- creación de vídeos educativos, incluyendo simulaciones cuando sea necesario
- configuración de entornos de aula y bases de datos para ser utilizados por los estudiantes

- creación o uso de material electrónico disponible basado en el aprendizaje de la investigación.

#### 4.3.2 AMT y PC

El PC consta de tres elementos principales:

- a. Algoritmos numéricos y no numéricos, así como software para modelado y simulación para la solución de problemas.
- b. Infraestructura informática para apoyar los problemas científicos y
- c. El desarrollo de sistemas de hardware y software, sistemas de internet, sistemas de gestión de datos, necesarios para la solución de problemas exigentes en computadoras (Grover & Pea, 2013).

Lo anterior indica la fuerte necesidad de incorporar métodos de aprendizaje electrónico mediados por tecnología en línea a la enseñanza de PC.

Uno de los objetivos cruciales en relación con la educación de los estudiantes es la adquisición de habilidades que les permitan resolver varios problemas. El educador, desarrollando la ética profesional, puede ser innovador mediante el uso de técnicas/prácticas de enseñanza con el fin de mejorar su trabajo educativo, confiando en hacer frente con éxito a los cambios en curso. Los obstáculos podrían aparecer por las creencias de ciertos educadores, es decir, sus ideas sobre la construcción del conocimiento, que determinan sus opciones de enseñanza, dirigen su comportamiento docente y configuran su enseñanza real.

Un punto clave para empezar es cambiar la forma de pensar sobre la enseñanza y el aprendizaje. Se sugiere que la reflexión continua y la autoevaluación del acto didáctico se ejerzan junto con el apoyo de la formación para el desarrollo profesional. El objetivo es producir y utilizar nuevos conocimientos pedagógicos con la ayuda de los propios educadores y la aplicación de su experiencia. El PC es algo nuevo que queremos implementar en el proceso educativo para mejorar la calidad de la enseñanza

y el aprendizaje de los estudiantes. Esto indica que el desarrollo profesional de todos los educadores, de todos los campos, es necesario para aplicar PC en nuestras áreas temáticas. La formación permanente de los educadores es reconocida hoy en día como una parte inherente a su carrera profesional y desarrollo.

En resumen, el progreso científico futuro en todas las sociedades depende del desarrollo del PC, por lo que los ciudadanos modernos deben prepararse para el futuro al estar equipados con habilidades de PC y familiarizarse con el desarrollo del PC y las ideas computacionales desde el principio, desde la educación básica. En años anteriores, las habilidades de los estudiantes involucraban escritura, lectura y aritmética. Hoy en día debemos añadir PC a esas habilidades. Esto es para mostrar la importancia de la preparación de los educadores para hacer frente a su trabajo complejo, que es la educación de los ciudadanos del futuro. PC incluye una gama de habilidades que todos los educadores están dispuestos a aprender a usar. Los primeros pasos se están dando con la incorporación del PC en los planes de estudio renovados en la educación obligatoria de muchos países.

En realidad, muchos países están invirtiendo simultáneamente en la entrega a largo plazo de ciencias de la computación (Wing, 2016). Sin embargo, es necesario eliminar el malentendido de que el PC solo está relacionada con el uso de computadoras. El desarrollo y la promoción del PC es una ventaja para el progreso científico y tecnológico que puede mejorar la vida de las personas y, por lo tanto, se hace necesario que se incorpore a la educación general. Los educadores, que serán llamados a incorporar el PC en este último, deben recibir apoyo con la debida preparación, para tener éxito en enfrentar la dinámica pedagógica del PC.

Todos los países europeos están tomando iniciativas para promover la lucha contra el terrorismo. Por ejemplo, en Grecia, los primeros pasos se dan con el concurso Bebras, ([www.bebas.org](http://www.bebas.org)), cuyo objetivo es sensibilizar a la comunidad educativa, con el fin de colmar las lagunas existentes en la

educación básica, en lo que respecta al PC. El objetivo del concurso antes mencionado es que los estudiantes se inspiren en diversos asuntos relacionados con la informática y el PC, que han sido adaptados según su edad por científicos informáticos de todo el mundo. La participación de los alumnos en el concurso es una experiencia de aprendizaje corta, entretenida y enriquecedora. La gran parte de los participantes de los alumnos, así como el de los educadores que han participado en la realización del concurso, muestra el reconocimiento del valor pedagógico del PC, junto con su intención de incorporarlo a su práctica diaria.

#### 4.4 «Formación del entrenador»

##### 4.4.1 ¿Por qué el pensamiento computacional?

Aprender PC no es lo mismo que la informática y no implica solo este campo. El PC se puede aplicar a todas las áreas científicas, así como en la vida diaria. Se centra principalmente en el desarrollo y resolución de problemas del mundo real. Utiliza una serie de habilidades y transforma problemas complejos en más simples.

Aquí parece apropiado referirse a las famosas 4 reglas de René Descartes, que es matemático y filósofo, descrito en su obra *Reglas para la dirección de la mente*:

- (1) no aceptar nada como verdadero que no sea evidente por sí mismo,
- (2) dividir los problemas en sus partes más simples,
- (3) resolver problemas procediendo de simple a complejo, y
- (4) volver a comprobar el razonamiento.

Usando la habilidad de eliminar detalles, hace que los problemas sean comprensibles y eso conduce a soluciones inmediatas. El mapa del ferrocarril subterráneo de Londres es un ejemplo, que incluye solo la información necesaria para determinar una ruta de una estación a otra (Cansu, S. K., & Cansu, F. K., 2019). Con la habilidad de descomposición

o análisis, el problema se divide en pequeños con elementos comprensibles. Con la generalización somos capaces de reconocer algunas de las cuestiones del problema como conocidas (Κούσης, 2017). Con el pensamiento algorítmico determinamos los pasos a seguir para encontrar soluciones a los problemas individuales, más simples, y con la automatización, creamos algoritmos para ser utilizados en otros problemas también (Cansu, S. K., & Cansu, F. K., 2019).

En el campo de la educación, se sugiere la creación de juegos, es decir, la aplicación de mecanismos que se dirigen a la movilización de los estudiantes debido a los motivos ofrecidos. De esta manera, aumenta la participación en actividades de aprendizaje, mientras que al mismo tiempo se sienten complacidos y satisfechos con su participación. El educador debe ser capaz de reconocer y utilizar la capacidad cognitivo-emocional del PC, que debe desarrollarse, de acuerdo con el objetivo de aprendizaje establecido. Por lo tanto, los educadores deben recibir motivación para preparar escenarios de aprendizaje atractivos, o incluso escenarios atractivos para usar, con el fin de ayudar a sus alumnos a resolver problemas con PC (Kuo, M.J., 2007).

El entorno tecnológico creado tendrá como objetivo activar e impulsar a los educadores hacia la creatividad. Todos los educadores deben tener acceso a los materiales educativos, las herramientas y las instrucciones creadas para que el PC se utilice correctamente.

Los estudiantes que están capacitados en el uso de PC adquieren habilidades importantes, como la confianza en sí mismos para hacer frente a problemas complejos, la tolerancia a la ambigüedad que puede existir en un problema, las habilidades de comunicación con sus compañeros mientras trabajan juntos para alcanzar una solución y habilidades de gestión de problemas, mientras que al mismo tiempo se vuelven más persistentes al tratar de resolver un problema (Kouzoukas, 2019).

#### 4.4.2 Estrategias educativas para formadores

Teniendo en cuenta todo lo anterior, surge una pregunta crítica: ¿De qué manera y con qué herramientas educativas se puede familiarizar un profesor con el PC? En primer lugar, los maestros tienen que ser persuadidos a utilizar el PC como un método muy útil y eficaz para sus lecciones. Para este propósito, los maestros necesitan entender los principios básicos del PC, y aprender a adoptarlos. Entonces se les debe proporcionar las razones para usarlo y para convencerse a sí mismos de su eficacia. Por último, deben ser proporcionados por los recursos adecuados y algunas herramientas que faciliten la aplicación del método de enseñanza en clase (Pinder, N., 2022).

Para que el PC se integre en diferentes áreas y materias de la educación obligatoria, es importante proporcionar a todos los educadores actuales y futuros un conocimiento suficiente de sus aspectos, habilidades y características, así como de cómo puede integrarse en las lecciones.

Jeannette Wing, de la Universidad de Columbia, trajo la idea del pensamiento computacional en un artículo que escribió en 2006. Ella dice: *“Informalmente, el pensamiento computacional describe la actividad mental en la formulación de un problema para admitir una solución computacional. La solución puede ser llevada a cabo por un ser humano o una máquina, o más generalmente, por combinaciones de seres humanos y máquinas»* (Alas, J., (2006). Esta es una gran declaración sobre el hecho de que PC se puede aplicar para lecciones en línea, pero no solo. Y hablando de «actividad mental» uno debe considerar la reproducción o aplicación de un algoritmo informático para analizar y finalmente llegar a la solución.

Hay tres razones principales por las que los educadores de todos los grados deben usar PC. En primer lugar, ayuda a los estudiantes a hacer frente a casi cualquier problema proporcionándoles un procedimiento paso a paso para llegar a una solución final, a través de una variedad de disciplinas, apalancamientos. Esta es una habilidad poderosa y un equipo valioso para sus vidas. En segundo lugar, exhibe la fuerza de la tecnología informática y, finalmente, complementa y

mejora el currículo escolar existente (Waterman, K., P. Goldsmith, L. & Pasquale, M. 2019).

Con el fin de incorporar PC en sus lecciones, los maestros necesitan saber cómo:

- Formular un problema determinado de una manera que nos permita utilizar una computadora y otras herramientas para ayudar a resolverlos.
- Organizar, analizar y representar lógicamente datos a través de abstracciones como modelos y simulación.
- Tener éxito en la automatización de soluciones a través del pensamiento algorítmico (una serie de pasos ordenados).
- Identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más eficiente y efectiva de pasos y recursos.
- Generalizar y transferir este proceso de resolución de problemas a una amplia variedad de problemas.

En la formación de la identidad compleja y múltiple de los estudiantes del siglo XXI, los profesores tienen un papel decisivo. Se necesita dedicación, capacitación y esfuerzo para crear una comunidad de educadores listos para transmitir a los estudiantes la esencia y la verdadera naturaleza del PC. Especialmente para los profesores a los que no se les ha enseñado las ciencias de la computación lo suficientemente bien como para poder incorporar conceptos clave en su enseñanza, por lo tanto, sus elecciones y grado de implicación dependen de su voluntad de formación y su entusiasmo hacia la materia.

El doble desafío de comprender el contenido y elegir la estrategia pedagógica adecuada para transferir conocimientos y habilidades en los estudiantes crea una situación bastante compleja para los profesores. Al igual que con los programas educativos AMT, parece que los educadores pueden beneficiarse significativamente a través de:

- participación en programas educativos

- talleres de capacitación adecuados
- las manos en los entrenamientos
- seminarios
- autoestudio
- participación en comunidades de aprendizaje y práctica, interesadas en el tema (Patton, K. Parker, M. 2017)

Además, se considera necesaria la cooperación de profesores con expertos en informática. Las percepciones de los maestros sobre la informática y su aplicación en diversas disciplinas pueden cambiar después de la formación (Yadav, A., Gretter, S., Good, J., & McLean, T. 2017).

Los programas educativos, la capacitación y los seminarios definitivamente ayudarán a los educadores a desarrollar una comprensión compartida de la enseñanza del PC en las escuelas, proporcionar el enfoque pedagógico para la enseñanza y ofrecer guías para la evaluación. Ayudarán a los profesores de preservicio a adquirir nuevas formas de pensar e incorporar TI a sus lecciones. (Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T., 2014). La formación de educadores en PC y métodos de enseñanza en línea podría organizarse en el marco de programas educativos interescolares o intraescolares de muchos países.

La participación en comunidades de aprendizaje y aplicación bajo el paraguas de AMT y PC en línea es sin ambigüedades el medio de autoentrenamiento más importante.

#### **4.4.3 Herramientas y recursos para el autoestudio**

Para animarse a utilizar el pensamiento computacional en clase, todos los profesores deben tener la posibilidad de tener a su disposición una serie de herramientas y recursos, con el fin de comprender en profundidad todos los aspectos de la aplicación del PC cuando enseñan sus asignaturas. Por supuesto, como se explicó anteriormente, la alfabetización digital es un requisito previo básico para esta última. Dado que el pensamiento computacional consiste en el desglose de un problema complejo a los más

simples, el reconocimiento de patrones y la creación del algoritmo (conjunto de instrucciones), está claro que un profesor podría usar el método en clase utilizando solo habilidades digitales básicas o incluso sin habilidades digitales. Por lo tanto, los maestros necesitan ser educados sobre las formas de integrar el pensamiento computacional a sus lecciones utilizando las disciplinas en las que se basa una computadora para resolver un problema o crear una actividad. Sin embargo, incluso si romper el problema y proporcionar un algoritmo podría basarse en una metodología que se puede crear sin el uso de las TIC, el reconocimiento y representación de patrones necesita fuertemente la ayuda de la tecnología digital para alentar a los estudiantes a usar su imaginación y reconocer situaciones similares desarrollando su forma crítica de pensar. El PC se puede aplicar a la mayoría de las asignaturas en la escuela, a partir de la educación primaria o incluso preescolar. Sin embargo, la primera consideración de un profesor recién llegado al mundo del pensamiento computacional debe ser una extensa búsqueda de literatura y auto-estudio sobre la definición y las disciplinas del PC.

- Los cursos en línea son organizados por las Universidades, centrándose en educar a los profesores — capacitar al formador — en diversas materias y grados, en el enfoque y las disciplinas de PC. Algunos de ellos proporcionan material gratuito para que los maestros usen en sus clases.

<p><b>I) UCX University of Canterbury</b> ofrece MOOCs, cursos gratuitos en línea sobre varios temas. El curso de pensamiento computacional proporciona conocimientos básicos de PC, así como herramientas para que los maestros enseñen pensamiento computacional, desde desconectado hasta enchufado con programación de computadoras. El curso se basa en el sitio web</p>	<p><a href="https://www.edx.org/course/teaching-computational-thinking">https://www.edx.org/course/teaching-computational-thinking</a></p>
---	--

<p>csunplugged.org, una colección de material didáctico gratuito para educadores</p>	
<p><b>II) Coursera</b> es una firma gratuita en red de proyectos, donde se dan referencias para una serie de formación online sobre pensamiento computacional.</p> <p>El MOOC «Problem Solving Using Computational Thinking» está organizado por la Universidad de Michigan.</p> <p>El curso se centra en las disciplinas de PC, “abstracción, identificación de problemas, descomposición, reconocimiento de patrones, algoritmos y soluciones de evaluación. Los estudiantes desconcertarán a través de algunos casos del mundo real que ilustran cómo el pensamiento computacional se puede utilizar para resolver problemas complejos. También completarán un proyecto que les permita aplicar el pensamiento computacional a una situación del mundo real».</p>	<p><a href="https://www.coursera.org/courseraplus">https://www.coursera.org/courseraplus</a></p> <p><a href="https://www.coursera.org/learn/comphinking?action">https://www.coursera.org/learn/comphinking?action</a></p> <p><a href="https://www.coursera.org/browse">https://www.coursera.org/browse</a></p>
<p>III) Sociedad Internacional de Tecnología en la Educación (ISTE), entre otras actividades, organiza cursos en línea. Uno de ellos se titula «<b>Introducción al Pensamiento Computacional para cada Educador</b>» y se centra en la enseñanza de cómo integrar PC a diferentes materias y niveles de grado.</p>	<p><a href="https://www.iste.org/areas-of-focus/computational-thinking-in-the-classroom">https://www.iste.org/areas-of-focus/computational-thinking-in-the-classroom</a></p>
<p>IV. Los cursos también son ofrecidos por <b>los proyectos KA1 Erasmus</b> de movilidad personal de profesores en el área de Tecnología de la Información y materias relacionadas. Debe alentarse a los profesores de las</p>	<p><a href="https://www.englishmatters.org/index.php/pages/view/erasmus_plus/erasmus?qclid">https://www.englishmatters.org/index.php/pages/view/erasmus_plus/erasmus?qclid</a></p>

<p>escuelas a aplicar a estos proyectos europeos en el marco de la educación STEM.</p>	
<p>v. <b>El Grupo de Investigación en Ciencias de la Computación de la Universidad de Adelaide</b> en Australia se ha asociado con Google para crear cursos introductorios para implementar el Currículo de Tecnologías Digitales de Australia y enseñar ciencias de la computación y pensamiento computacional en los niveles primario y secundario, explícitamente vinculados al plan de estudios australiano.</p>	<p>(plazo de <a href="https://csdigitaltech.appspot.com">https://csdigitaltech.appspot.com</a></p>

- Las plataformas digitales, las redes y los sitios web también son una poderosa herramienta para la autoformación y el estudio de los conceptos de PC, a través de la información proporcionada, las comunidades de aprendizaje y actuación, los recursos y el material didáctico. Por lo general, son proporcionados por organizaciones de apoyo como CSTA, ISTE y la Asociación Nacional de Profesores de Ciencias (NSTA), que también están desarrollando y compartiendo herramientas y recursos en línea para maestros actuales y futuros.

<p><b>CAS</b> (Computing at Schools) es una red internacional de profesores, académicos y profesionales que participan en ciencias de la computación y proporcionan recursos y material didáctico.</p> <p>CAS ofrece una interesante guía gratuita para profesores involucrados en el Pensamiento Computacional.</p>	<p><a href="https://www.computingat-school.org.uk/teaching-resources/2014/june/cas-computational-thinking-a-guide-for-teachers">https://www.computingat-school.org.uk/teaching-resources/2014/june/cas-computational-thinking-a-guide-for-teachers</a></p> <p><a href="https://www.computingat-school.org.uk/media/ajlyssi/150818computationalthinking.pdf">https://www.computingat-school.org.uk/media/ajlyssi/150818computationalthinking.pdf</a></p>
<p>ii) <b>BSB</b> Education (Build Something Different) es una comunidad de expertos que apoyan a maestros, padres</p>	<p><a href="https://bsd.education/l/computational-thinking/">https://bsd.education/l/computational-thinking/</a></p>

<p>y estudiantes y les proporcionan habilidades digitales. Es una fundación pedagógica que, entre otros, proporciona lecciones en línea.</p> <p>BSB proporciona a petición un libro de trabajo sobre PC con actividades para aplicar en clase</p>	
<p>iii) <b>Pensamiento computacional en K-12 La organización ISTE</b> ( International Society for Technology Education) en colaboración con la Asociación de Profesores de Ciencias de la Computación(CSTA) creó un breve manual sobre educación K-12. El libro describe las disciplinas básicas del PC, las habilidades necesarias para enseñar utilizando las disciplinas de PC y proporciona algunas experiencias de aprendizaje en varios temas.</p>	<p><a href="https://cdn.iste.org/www-root/2020-10/ISTE_CT_Teacher_Resources_2ed.pdf">https://cdn.iste.org/www-root/2020-10/ISTE CT T eacher Resources 2ed.p df</a></p>
<p>iv) <b>NEOBLOG</b> es una plataforma de recursos que apoya a los profesores que proporcionan herramientas y métodos de enseñanza en línea probados. Proporciona conocimientos básicos sobre conceptos de PC y algunas herramientas digitales que podrían ayudar a los maestros a aplicar lecciones de PC en clase: Scratch, Kodable, MineKraft y otros.</p>	<p><a href="https://blog.neolms.com/6-digital-tools-that-encourage-computational-thinking/">https://blog.neolms.com/ 6-digital-tools-that-enco urage-computational-thi nking/</a></p> <p><a href="https://scratch.mit.edu/">https://scratch.mit.edu/</a> <a href="https://www.kodable.com/">https://www.kodable.com /</a> <a href="https://education.minecraft.net/">https://education.minecr aft.net/</a></p>
<p>v) <b>CC Unplugged</b> es una colección de materiales didácticos gratuitos sobre ciencias de la computación y proporciona capacitación a educadores en varias materias de Tecnología de la Información.</p>	<p><a href="https://www.csunplugged.org/en/">https://www.csunplugged .org/en/</a></p>
<p>vi) El sitio web <b>Exploring Computational Thinking de Google</b> proporciona más de 130 planes de lecciones</p>	<p><a href="http://q.co/exploringCT">http://q.co/exploringCT</a></p>

<p>y programas de muestra alineados con los estándares internacionales de educación; una colección de videos que demuestran cómo se utilizan los conceptos de pensamiento computacional en la resolución de problemas del mundo real, así como un curso en línea «Pensamiento computacional para educadores» (<a href="http://q.co/computationalthinking">http://q.co/computationalthinking</a>).</p>	
<p>vii) <b>TPACK</b>: Marco de conocimiento de contenidos pedagógicos tecnológicos. El marco de seguimiento combina tres tipos de conocimiento, conocimiento tecnológico (TK), conocimiento pedagógico (PK) y conocimiento de contenido (CK). La teoría de la pista debe adaptarse de los educadores para mejorar su desarrollo profesional.</p>	<p><a href="https://educationaltechnology.net/technological-pedagogical-content-knowledge-tpack-framework/">https://educationaltechnology.net/technological-pedagogical-content-knowledge-tpack-framework/</a></p>
<p>viii) <b>Iniciativas de Pensamiento Computacional</b>. Esta página web ofrece programas y recursos con el objetivo de desarrollar y cultivar el pensamiento computacional para educadores y estudiantes.</p>	<p><a href="https://www.computationinitiative.org/">https://www.computationinitiative.org/</a></p>

## Referencias

Adsit N.J. (2004): Technology Mediated Professional Development Programs. American Association of Colleges for Teachers Education.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads*, 2(1), 48-54

Borthwick A.C. & Hansen R. (2017) Digital Literacy in Teacher Education: Are Teacher Educators Competent?, *Journal of Digital Learning in Teacher Education*, 33:2, 46-48, DOI: 10.1080/21532974.2017.1291249

Cansu, S.K. & Cansu F.K., (2019): An Overview of Computational Thinking. International Journal of Computer Science Education in Schools, April 2019, Vol. 3, No. 1, ISSN 2513-8359

Denning, P. J. (2017). Remaining trouble spots with computational thinking Communications of the ACM, 60(6), 33-39

Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015). Computing education in K-12 schools: A review of the literature. In 2015 IEEE Global Engineering Education Conference (EDUCON) (pp. 543-551). IEEE

Grover, S. & Pea, R., Computational Thinking in K-12. Educational Researcher, Vol. 42 No. 1, pp. 38–43. doi: 10.3102/0013189X12463051

Heintz, F., Mannila, L., & Farnqvist, T. (2016). A Review of Models for Introducing Computational Thinking, Computer Science and Computing in K–12 Education. Frontiers in Education Conference (FIE) (pp 1–9). IEEE. doi:10.1109/fie.2016.7757410

Falloon, G. (2020). From digital literacy to digital competence: the teacher digital competency (TDC) framework Education Tech Research Dev. <https://doi.org/10.1007/s11423-020-09767-4>

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? Computers in Human Behavior, 41, 51–61. doi:10.1016/j.chb.2014.09.012

Kakavas, P., & Ugolini, F. C. (2019). Computational thinking in primary education: A systematic literature review. Research on Education and Media, 11(2), 64-94.

P. Knight (2002), A systemic approach to professional development: learning as practice. Teaching and Teacher Education 18 (2002) 229–241.

Knight, S.L., Wiseman D. (2005): Professional Development for Teachers of Diverse Students. Journal of Education for Students placed at Risk: A Summary of the Research, 10(4), 387–405

Kouzoukas, G., (2019): Thesis, Design and evaluation of an introductory teacher training workshop to understand the concept of Computational Thinking

Kuo, M.J. (2007): How does an online game based learning environment promote students' intrinsic motivation for learning natural science and how does it affect their learning outcomes? The First IEEE International

Workshop on Digital Game and Intelligent Toy Enhanced Learning, Jhongli City, Taiwan.

Magno, C. The role of metacognitive skills in developing critical thinking. *Metacognition Learning* 5, 137–156 (2010).  
<https://doi.org/10.1007/s11409-010-9054-4>

Mason, S. L., & Rich, P. J. (2019). Preparing elementary school teachers to teach computing, coding, and computational thinking. *Contemporary Issues in Technology and Teacher Education*, 19(4), 790-824

Oliver, R. & Herrington, J. (2003): Exploring Technology-Mediated Learning from a Pedagogical Perspective, *Interactive Learning Environments*, 11:2, 111-126

Pala, F. K., & Mıhçı Türker, P. (2021). The effects of different programming trainings on the computational thinking skills. *Interactive Learning Environments*, 29(7), 1090-1100.

Patton, K. & Parker, M. : Teacher education communities of practice: More than a culture of collaboration. *Teaching and Teacher Education* Volume 67, October 2017, Pages 351-360, <https://doi.org/10.1016/j.tate.2017.06.013>

Pinder, N., (2022): ISTE  
: <https://www.iste.org/explore/computational-thinking/why-you-should-integrate-computational-thinking-your-curriculum>.

Saidin, N. D., F. Khalid, R. Martin, Y. Kuppusamy, and N. A. Munusamy. Benefits and Challenges of Applying Computational Thinking in Education. *International Journal of Information and Education Technology* 11, no. 5 (2021): 248-254

Scherer, R. (2016). Learning from the past—the need for empirical evidence on the transfer effects of computer programming skills. *Frontiers in Psychology*, 7, 1390

Shailaja, J., & Sridaran, R. (2015). Computational thinking the intellectual thinking for the 21st century. *International Journal of Advanced Networking & Applications*, 7, 39-46

Selby, C., & Woollard, J. (2014). "Refining an understanding of computational thinking

Suskind, D. (2016): Teacher as Researcher: The Ultimate Professional Development. <https://www.edutopia.org/>

Synchronous tools and how to use them. Center for Academic Innovation, (2020)

<https://onlineteaching.umich.edu/synchronous-tools-and-how-to-use-them/>

UNICEF Greece (2022): A day at school during COVID-19

<https://www.unicef.org/greece/en/stories/day-school-during-covid-19>

Waterman, K., P. Goldsmith, L. & Pasquale, M. (2019): Integrating Computational Thinking into Elementary Science Curriculum: an Examination of Activities that Support Students' Computational Thinking in the Service of Disciplinary Learning. *Journal of Science Education and Technology* <https://doi.org/10.1007/s10956-019-09801-y>

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, 14(1), 1-16

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*, 60(6), 565-568

Yadav, A., Gretter, S., Good, J., & McLean, T. (2017). Computational Thinking in Teacher Education. *Emerging Research, Practice, and Policy on Computational Thinking*, 205–220. doi:10.1007/978-3-319-52691-1\_13

Young, L. & Paterson, B. (2007): *Developing a Student-Centered Learning Environment*. Lippincott Williams & Wilkins, p.5

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35

Wing, J., (2016). Computational Thinking and thinking about Computers. *Phil. Trans. R. Soc. A* (2008) 366, 3717–3725. doi:10.1098/rsta.2008.0118.

