



Ciencia Latina
Internacional

Ciencia Latina Revista Científica Multidisciplinar, Ciudad de México, México.
ISSN 2707-2207 / ISSN 2707-2215 (en línea), mayo-junio 2024,
Volumen 8, Número 3.

https://doi.org/10.37811/cl_rcm.v8i3

**DESARROLLO DE COMPETENCIAS DE
EGRESO EN ESTUDIANTES DEL NMS
MEDIANTE LA CODIFICACIÓN
COMPUTACIONAL**

**DEVELOPMENT OF GRADUATION COMPETENCIES IN NMS
STUDENTS THROUGH COMPUTATIONAL CODING**

María Guadalupe Aguilar Espinosa

Universidad de Guanajuato, México

Rocío Rubio Rivera

Universidad de Guanajuato, México

Mónica Mondelo Villaseñor

Universidad de Guanajuato, México

Hilda Lucia Cisneros López

Universidad de Guanajuato, México

DOI: https://doi.org/10.37811/cl_rcm.v8i3.13420

Desarrollo de Competencias de Egreso en Estudiantes del NMS mediante la Codificación Computacional

María Guadalupe Aguilar Espinosa¹

mgaguilera@ugto.mx

<https://orcid.org/0009-0007-7561-9421>

Universidad de Guanajuato

Escuela de Nivel Medio Superior de Salvatierra
México

Rocío Rubio Rivera

r.rubio@ugto.mx

<https://orcid.org/0009-0005-8194-8681>

Universidad de Guanajuato

Escuela de Nivel Medio Superior de Salvatierra
México

Mónica Mondelo Villaseñor

m.mondelo@ugto.mx

<https://orcid.org/0000-0002-8485-055X>

Universidad de Guanajuato

Escuela de Nivel Medio Superior de León
México

Hilda Lucia Cisneros López

docentenms@gmail.com

<https://orcid.org/0000-0003-3245-0279>

Universidad de Guanajuato

Escuela de Nivel Medio Superior de Salvatierra
México

RESUMEN

El presente artículo presenta los resultados obtenidos en la investigación realizada en la Escuela de Nivel Medio Superior de Salvatierra, en la que se midió el impacto que tiene el uso de la programación computacional en el desarrollo de habilidades de orden superior en los estudiantes, para tal efecto se siguió una ruta cuantitativa, aplicando como instrumento un cuestionario con escala de Likert a una muestra intencional sobre los estudiantes que desde el nivel de estudios básicos comenzaron con la programación ya sea de manera formal o a través de un aprendizaje social. Los resultados refieren un reconocimiento importante por parte de los estudiantes sobre las habilidades críticas, analíticas, reflexivas, creativas, entre otras, que les permiten generar profundos procesos de pensamiento en cualquier contexto de su vida, y por lo tanto les permiten académicamente contar con un desenvolvimiento distinto al resto de sus compañeros, dándoles ventajas competitivas que se reflejan en su vida académica. Lo anterior destaca que los docentes debemos incluir en nuestros planes y programas de estudio proyectos transversales que incluyan la programación computacional para resolver problemas de la vida cotidiana y además despierten o fomenten las habilidades de pensamiento.

Palabras claves: estudiante, habilidades, egreso, programación

¹ Autor principal

Correspondencia: mgaguilera@ugto.mx

Development of Graduation Competencies in NMS Students through Computational Coding

ABSTRACT

This article presents the results obtained in the research carried out at the Salvatierra High School, in which the impact of the use of computer programming on the development of higher order skills in students was measured, for such purpose. For this purpose, a quantitative route was followed, applying Likert scale instruments and questionnaires to an intentional sample of students who began programming from the basic education level, either formally or through social learning. The results refer to an important recognition on the part of the students about the critical, analytical, reflective, creative skills, among others, that allow them to generate deep thought processes in any context of their lives, and therefore allow them to have an academic development different from the rest of their classmates, giving them competitive advantages that are reflected in their academic life. The foregoing highlights that teachers must include in our study plans and programs transversal projects that include computer programming to solve problems of daily life and also awaken or promote thinking skills.

Keywords: student, skills, egress, programming

Artículo recibido 24 julio 2024
Aceptado para publicación: 27 agosto 2024



INTRODUCCIÓN

En el ámbito de la Educación Media Superior, el desarrollo de competencias de egreso es fundamental para garantizar que los estudiantes estén preparados para enfrentar los desafíos académicos y profesionales que se les presentarán en el futuro. La programación computacional emerge como una herramienta innovadora y efectiva para alcanzar este objetivo. Al integrar la programación en el currículo, los estudiantes no solo adquieren habilidades técnicas en programación, sino que también desarrollan competencias clave como el pensamiento lógico, la resolución de problemas y la creatividad. Estas competencias son esenciales para su éxito en una sociedad cada vez más dominada por la tecnología, lo que puede potenciar las competencias de egreso de los estudiantes, preparándolos de manera integral para su futuro académico y profesional.

Para Perrenoud (2004) la competencia es “la capacidad de actuar eficazmente en un tipo definido de situación, capacidad que se apoya en conocimientos, pero no se reduce a ellos”. Para la OECD (2005) las competencias son “la capacidad de responder a demandas complejas y llevar a cabo tareas diversas de forma adecuada”. Aquí se destaca la adecuación y efectividad en la respuesta a demandas complejas. En este sentido ambos coinciden en la idea de que una competencia es una capacidad que integra conocimientos, habilidades y actitudes que permiten a una persona actuar de manera eficaz y adaptativa en diferentes contextos

De acuerdo con Anderson (1983) existe diferencia entre el conocimiento declarativo y el conocimiento procedimental. Por una parte, el conocimiento declarativo incluye información fáctica y verbal, recuerdos basados en signos, sentidos y en la percepción. Adicionalmente, abarca hechos y conceptos relacionados con eventos de la vida diaria hasta conocimientos conceptuales organizados e interrelacionados. En contraste, el conocimiento procedimental, comprende la habilidad de monitorear los propios procesos cognitivos (metacognición), así como la capacidad automatizada de transformar y organizar nuevos métodos para resolver problemas.

Para Montano (2021) los procesos de pensamiento son conjuntos de acciones relacionadas con la elaboración de conocimientos a partir de estímulos internos y externos, para procesar información. De acuerdo con este autor, el aprendizaje está relacionado con los procesos de observación, comparación, clasificación, análisis o descripción los cuales se activan para construir conocimientos nuevos y a través



de los cuales se construye la capacidad de pensar, procesar información o tener actitud crítica. Los procesos básicos de pensamiento se relacionan con la forma en la cual el individuo percibe la realidad que le rodea, mientras que los procesos de pensamiento integradores surgen una vez que se han completado los procesos básicos de pensamiento.

Resulta conveniente distinguir entre las habilidades cognitivas que se emplean para ejecutar el proceso de codificación, almacenamiento, recuperación y transformación de la información, y los procesos de orden superior (metacognitivos o ejecutivos) necesarios para poner en marcha los procesos de orden inferior y para monitorear el resultado de las transformaciones y respuestas generadas por dichos procesos. Los términos que se usan para distinguir entre las habilidades de pensamiento de orden inferior y las de orden superior varían notablemente, pero hay cierto consenso acerca del hecho de que estas habilidades metacognitivas están entre las competencias mentales más transferibles (Warner y Sternberg, 1984).

El concepto de habilidades cognitivas de orden superior se relaciona directamente con procesos mentales necesarios en el análisis de actividades complejas. A finales del siglo XX, Bloom estableció cuáles eran estos procesos motivo por el cual, la taxonomía de Bloom se convirtió en referencia en el mundo educativo. En un aula tradicional, comúnmente se presentan tres momentos (explicación, práctica y corrección), lo cual únicamente fomenta el desarrollo de tres habilidades de pensamiento cognitivo: recordar, comprender y aplicar. Uno de los objetivos que se persigue con la introducción de las TIC en los procesos de enseñanza y aprendizaje, es desarrollar el nivel superior del pensamiento cognitivo (Fernández, 2018).

La enseñanza y aprendizaje de la programación busca la formación y desarrollo de habilidades en los alumnos para favorecer la resolución de problemas en el ámbito escolar, profesional o de la vida cotidiana (Díaz, Fierro y Muñoz, 2018).

Para Cuny, Snyder y Wing (2010) el pensamiento computacional es “el proceso de pensamiento que interviene en la formulación de los problemas y sus soluciones, de manera que las soluciones se representen de forma que pueda ser realizada por un **procesador de información**”. El valor de este concepto radica en la aplicación de este concepto en la representación de la solución de un problema a



través de secuencias de instrucciones y algoritmos. De esta forma se utilizaría el pensamiento abstracto para identificar aspectos relevantes y una secuencia de procesos para desarrollar un modelo.

Desde esta perspectiva, este concepto es trasladable y de gran relevancia en el ámbito educativo en donde la tecnología además de impulsar el aprendizaje de los alumnos para fomentar su interés en áreas del conocimiento relacionadas con la ciencia, tecnología, ingeniería, matemáticas e informática. La inclusión del pensamiento computacional en los programas educativos fomenta en los alumnos el desarrollo de competencias que les permitirá pasar de ser consumidores a productores de tecnología.

La implementación en los procesos de enseñanza y aprendizaje de actividades que promuevan el desarrollo del pensamiento computacional genera las siguientes ventajas para el aprendizaje de los alumnos (UNIR, 2021):

- Estimula la **creatividad**.
- Favorece la capacidad de **razonamiento** y de pensamiento crítico.
- Desarrolla y refuerza las **habilidades numéricas y lingüísticas**.
- Fomenta las habilidades **de liderazgo** y trabajo en equipo.

Sin embargo, el desafío es contar con docentes capaces de incorporar en su práctica docente actividades relacionadas con el pensamiento computacional. A nivel mundial la implementación en instituciones educativas de actividades relacionadas con el pensamiento computacional es una tendencia, pues se considera que favorece la innovación y se asocia con una inversión a largo plazo para la estructura económica y social de un país.

METODOLOGÍA

La ruta de investigación definida en el presente trabajo fue cuantitativa, definiendo una muestra intensional con los estudiantes inscritos en la especialidad de Físico-Matemáticas de la Escuela de Nivel Medio Superior de Salvatierra, Guanajuato, México, esta muestra consta de 58 estudiantes quienes cursan la Unidad de Aprendizaje de Lenguajes de Programación

Tipo de Investigación

Exploratorio y Descriptivo: Permitirá comprender las habilidades que adquieren los estudiantes cuando usan la programación computacional.

Diseño



Transversal: Se recolectaron datos en un solo punto en el tiempo.

Fenomenológico: Se busca comprender las experiencias y percepciones de los estudiantes en la clase de Lenguajes de Programación.

Población de Estudio

Estudiantes que cursen el quinto semestre de la especialidad Físico-Matemáticas y estén inscritos en la UDA de Lenguajes de programación.

Muestra

Se selecciona una muestra representativa de 58 estudiantes

Muestreo Aleatorio Simple : Para asegurar representación de diferentes estudiantes que durante la clase están participando e interactuando

Técnicas de Recolección de Datos

Cuestionario con escala tipo likert para evaluar percepciones y estado socioemocional

Instrumentos de Recolección de Datos

Cuestionario de encuesta con escala tipo likert

Consideraciones Éticas

Se obtiene el consentimiento informado de los participantes

Se garantiza la confidencialidad y anonimato de los datos

Se respetan los principios éticos de la investigación científica

Criterios de Inclusion y Exclución

Inclusión: Estudiantes inscritos en la UDA Lenguajes de programación que cursan en quinto semestre.

Exclusión: Estudiantes que no deseen participar o que no otorguen su consentimiento informado

Bajo este contexto se define la hipótesis “Los estudiantes que resuelven problemas a través de la programación computacional desarrollan habilidades de razonamiento, pensamiento crítico, numéricas,



lingüísticas, entre otras, llamadas habilidades de orden superior. Por tal motivo se desprenden los siguientes objetivos:

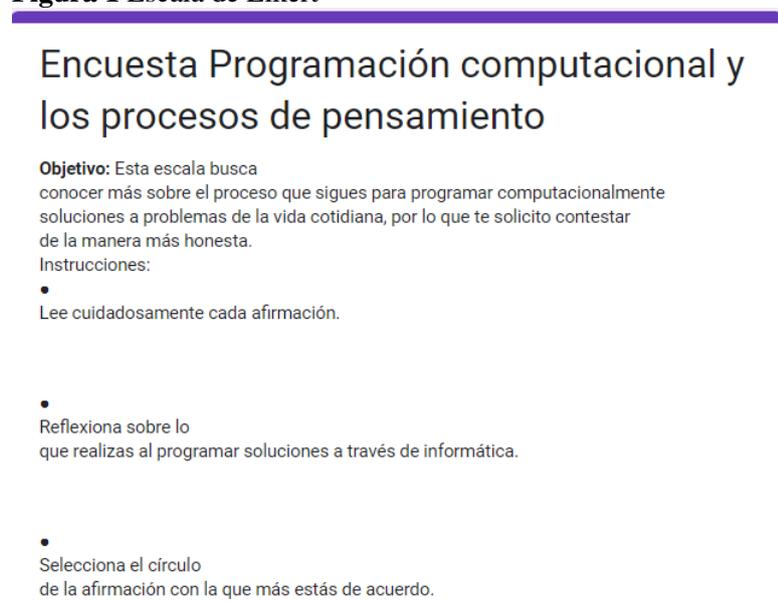
Medir el impacto que tiene el uso de la programación computacional en las habilidades de orden superior en los estudiantes

Objetivos particulares

Identificar estudiantes de quinto semestre que utilicen la programación computacional para resolver problemas

Medir a través de una escala de Likert con un conjunto de ítems validado por el cuerpo de investigadores, el impacto que tiene el uso de la programación en los estudiantes. Para esto se trabajó con Google Forms (Figura 1)

Figura 1 Escala de Likert



Encuesta Programación computacional y los procesos de pensamiento

Objetivo: Esta escala busca conocer más sobre el proceso que sigues para programar computacionalmente soluciones a problemas de la vida cotidiana, por lo que te solicito contestar de la manera más honesta.

Instrucciones:

- Lee cuidadosamente cada afirmación.
- Reflexiona sobre lo que realizas al programar soluciones a través de informática.
- Selecciona el círculo de la afirmación con la que más estás de acuerdo.

RESULTADOS Y DISCUSIÓN

A continuación, se resaltan los resultados más significativos tanto de la escala de Likert como del cuestionario que nos permitirá concluir acerca del impacto que tiene el uso de la programación computacional en el nivel medio superior hacia las habilidades superiores mentales en los estudiantes, tan importantes de ser desarrolladas a temprana edad.

Las siguientes figuras plasman los resultados más sobresalientes al momento de realizar tanto la escala de Likert como el cuestionario aplicado a toda la muestra poblacional definida en la metodología sin

distingo de la forma en que se ha adquirido el conocimiento sobre programación computacional, es decir un aprendizaje formal o un aprendizaje social.

Figura 2 Análisis de los problemas

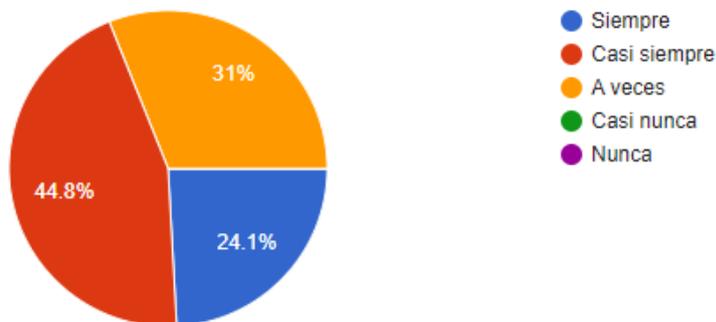
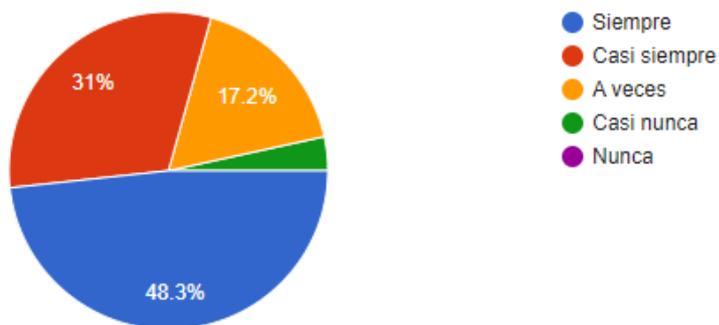


Figura 3 Comprensión lectora del problema



En la figura 2, refieren que más del 72% de estudiantes mencionan que para poder dar solución a un problema lo descomponen en pequeños problemas, lo que señalan le genera una perspectiva amplia de los elementos que conforman el problema. Como observamos en la figura 3, más del 79% de estudiantes refieren que el leer varias veces un problema los lleva a poder comprender mejor los elementos del

problema y cómo pudieran resolverlo. En la encuesta se habla sobre procesos de comprensión lectora, lo que coadyuva en las materias como física, química, inglés, español, entre otras.

Figura 4 Uso de metodologías

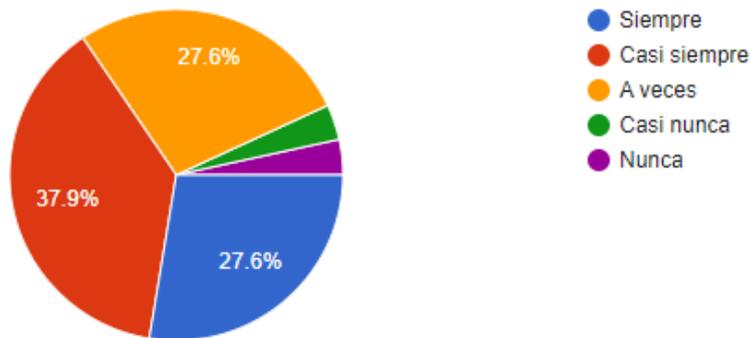
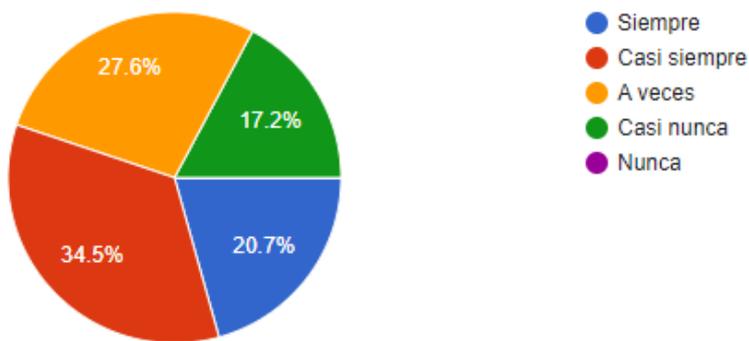


Figura 5 Uso de algoritmos



De acuerdo con la figura 4, el 65% de los estudiantes refieren emplear una metodología computacional para poder analizar y modelar el problema antes de programar alguna solución. Como podemos observar en la figura 5, el 55% de estudiantes hacen uso de algoritmos como metodología de análisis de problemas, seguido de diagramas de flujo y por último el pseudocódigo. En el cuestionario se aborda el

uso de metodologías para lograr proceso de análisis en la resolución de problemas, siendo apremiante en materias como matemáticas y ciencias experimentales.

Figura 6 Validación de algoritmos

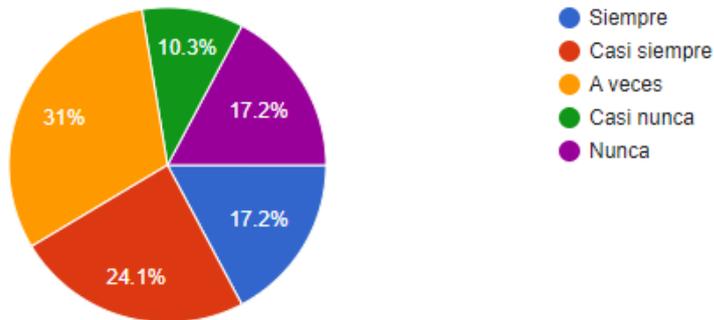
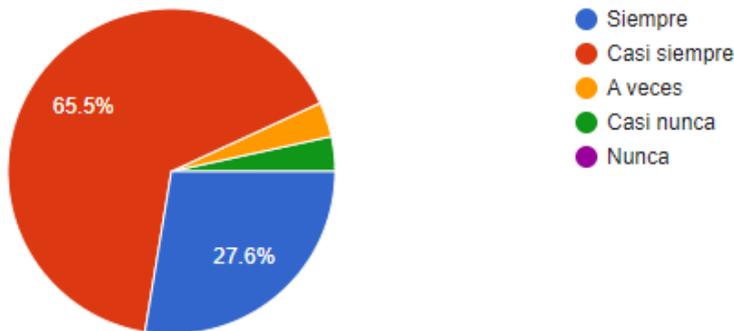
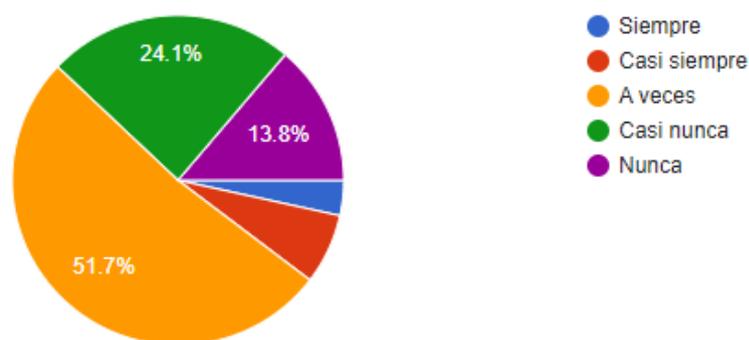


Figura 7 Consideración de posibles escenarios



De acuerdo con la figura 6. El 41% de los estudiantes refieren que antes de comenzar a programar una solución hacen uso de la validación previa del algoritmo, es decir, verifican que la propuesta planteada tendrá éxito validando cada uno de los procesos definidos en el algoritmo. Así la figura 7 refiere que más del 90% de los estudiantes mencionan que al analizar la solución consideran todos los posibles escenarios que pueden ocurrir en el proceso. En el cuestionario se aborda la consideración de diferentes rutas o caminos que puede tener un problema, lo que es de gran utilidad en materias como ciencias experimentales.

Figura 8 Lógica al ejecutar un programa



Al menos el 50% de los estudiantes de acuerdo con la figura 8, mencionan tener problemas de lógica al ejecutar una solución programada por ellos mismos. Aunque en la encuesta refieren que al inicio de la programación el fallo de la lógica era por encima del 50% y el otro porcentaje a problemas de sintaxis con el paso del tiempo han ido mejorando, teniendo mucho menos errores de lógica, ya que refieren ir mejorando con la práctica.

Algo también importante en la encuesta fue reconocer que su desempeño académico es aceptable, principalmente refieren tener menos problemas de aprendizaje en las unidades de aprendizaje de matemáticas, física, química y lenguajes de comunicación principalmente, además de que observan diversas oportunidades para incursionar como olimpiadas, concursos, Expociencias, entre otros.

CONCLUSIÓN

El uso de la programación computacional a cualquier edad de la vida de una persona es de gran utilidad, y más si se fomenta desde la niñez a lo que ahora se llama programación temprana. El uso de las tecnologías es otro factor importante para considerar, por las generaciones de estudiantes que actualmente se encuentran en las aulas llamados nativos digitales, los cuales buscan resolver problemáticas haciendo uso de la tecnología. Por lo que, derivado de la preocupación por lograr perfiles de egreso en los estudiantes de nivel medio superior, principalmente en las habilidades lógico-matemáticas, pensamiento crítico, reflexivo, iniciativa, creatividad, entre otras, que caen en lo que se denomina las habilidades superiores de pensamiento, es necesario voltear a ver las tecnologías que coadyuvan en gran medida a desarrollar los requerimientos de egreso en los estudiantes.

Los resultados del estudio desde la perspectiva de los estudiantes profundizan en la importancia que tiene el uso de la programación computacional para el desarrollo de dichas habilidad y de igual forma la necesidad de incorporarla en cualquier unidad de aprendizaje, no como una materia más sino como un aprendizaje transversal que apoye en la resolución de problemas en cualquier entorno que se presenten, y que de manera paralela enriquezca los perfiles deseados.

Así mismo se observa en los resultados la percepción que tienen los estudiantes en su formación académica respecto a la programación es muy notoria, ya que observan una manera distinta en plantear soluciones a los problemas que se les presentan, generando procesos complejos de analisis que derivan en el logro de distintas habilidades de orden superior como las ya mencionadas, apotando con ello estudiantes más participativos, creativos, emprendedores, reflexivos, generando con ello procesos de enseñanza aprendizaje más profundo y con una transversalidad significativa. Tal como señalaba Steve Jobs el programar una computadora te enseña a pensar.

Definitivamente una recomendación primordial sería incorporar en un curriculum ampliado desde niveles básicos de primaria y secundaria el uso de la programación computacional, ya que esto fomentaría el impacto de su uso a lo largo del trayecto académico del estudiante, hasta llegar a un trayecto formativo de niveles educativos superiores y en una inseercción en la vida laboral aportando al desarrollo de nuestro país. En fechas recientes los cambios señalados en la reforma educativa para nivel medio superior, señalan la necesidad de incorporar materias relacionadas con la programación computacional desde los primeros semestres sin importar la especialidad que se este cursando, lo que es de llamar la atención que el impacto que espera la Secretaría debe sustantivo en esta propuesta.

REFERENCIAS BIBLIOGRÁFICAS

Anderson, J. R. (1983). The architecture of cognition. Cambridge: Harvard University Press.

Fernández, P. (06-03-2018). Las Habilidades Cognitivas de Orden Superior. Actividades prácticas para su desarrollo en el aula. Blog: Un Aula para todos. TAC, competencia digital, inclusión y mucho más. <https://unaulaparatodos.wordpress.com/2018/06/03/las-habilidades-cognitivas-de-orden-superior-actividades-practicas-para-su-desarrollo-en-el-aula/>

Montaño, Joaquín. (27 de abril de 2021). **Procesos del pensamiento: qué son y cuáles son (con ejemplos)**. Lifeder. Recuperado de <https://www.lifeder.com/procesos-del-pensamiento/>



OECD. (2005). The definition and selection of key competencies: Executive summary. OECD Publishing.

Perrenoud, P. (2004). Desarrollar la práctica reflexiva en el oficio de enseñar: Profesionalización y razón pedagógica. Graó.

UNIR. (2021) ¿Qué es el pensamiento computacional?. UNIR REVISTA. Recuperado de <https://www.unir.net/educacion/revista/pensamiento-computacional/>

Wagner, R. K. y Sternberg, R. J. (1984). Alternative conceptions of intelligence and their implications for education. Review of Educational Research. 54 pp. 179-224.



Apéndice

Tabla 1 Escala de Likert

PREGUNTA	Siempre	Casi siempre	A veces	Casi nunca	Nunca
Un problema lo descompone en pedazos pequeños para poder solucionarlo					
Lee el problema en repetidas ocasiones hasta comprenderlo					
Utiliza una metodología para la resolución de problemas					
Emplea algoritmos para la solución de problemas de cualquier índole					
Emplea diagramas de flujo para la solución de problemas de cualquier índole					
Emplea pseudocódigos para la solución de problemas					
Antes de codificar la solución en un lenguaje de programación realiza la validación del algoritmo					
Antes de codificar la solución en un lenguaje de programación realiza un pseudocódigo					
Considera todos los posibles escenarios en una solución de un problema					
Las soluciones planteadas presentan errores en la secuencia lógica de las instrucciones					
Emplea estructuras de control condicionales en la solución de problemas					
Emplea estructuras de control repetitivas en la solución de problemas					
Modelo a través del dibujo o imaginación el problema antes de plantear una solución					
La codificación de la solución en el lenguaje de programación genera errores de sintaxis al ejecutarla					
La codificación de la solución en el lenguaje de programación genera errores de lógica al ejecutarla					

