

WEBINAR “CÓMO ENSEÑAR PROGRAMACIÓN Y PENSAMIENTO COMPUTACIONAL?”

Viernes 8 julio 2022, 09:30 hrs.

Organizan:

- [Observatorio de Prácticas Educativas Digitales OPED UC.](#)
- [Centro de Estudios de Políticas y Prácticas en Educación CEPPE UC.](#)

Facultad de Educación, Pontificia Universidad Católica de Chile.

PONENCIA: FRANCISCO VICO

[Científico e ingeniero, catedrático de inteligencia artificial en la Universidad de Málaga, España.](#)

Empezaré contándoles una historia: Cuando yo tenía 18 años, mi padre me llevó a un polígono industrial a las afueras de Málaga; me pidió que me sentara al volante y me dio unas explicaciones básicas: cómo poner punto muerto, cómo arrancar... Unas horas después me había hecho con el control de aquel viejo auto. Me resultó muy curioso que cuando le preguntaba, solía responder de forma ambigua: ¿cuándo cambiar de marcha?, el motor te lo dirá; ¿cuándo debo hacer rampa con el embrague?, si no lo haces, se cala; o ¿en qué marcha se sube esta rampa?, en la misma que se baja... Es decir, comprendí que la máquina me enseñaría cómo hacerlo. Mi padre era uno de los mejores conductores que he conocido, sin embargo, sólo necesité que me guiara un poco y me quitase el miedo a ocupar el asiento del conductor. El resto sólo podía aprenderlo por mí mismo.

En realidad, esto no era en absoluto sorprendente. Dos años antes había llegado a mis manos un ordenador y en un mes estaba haciendo código. Parte del mérito fue del lenguaje de programación, muy sencillo de aprender, y otra parte del manual, diseñado con esmero para que cada concepto fuera asimilado en la secuencia y forma adecuadas. Hoy, ese manual está entre los libros de mi despacho, para recordarme cómo se debe enseñar a programar.

Desde aquel boom de la programación, han pasado cuatro décadas sin que el ordenador termine de encontrar su lugar en las aulas. BBC Learning produjo dos series de Computer Literacy, en los 80 y en 2012, sin continuidad. Se ha dotado a los centros y su alumnado de portátiles y tabletas, en ocasiones con un coste muy elevado. Se ha desarrollado una teoría pedagógica en torno a ello, el pensamiento computacional, que describe competencias deseables para el estudiante. Ha florecido una industria de robótica educativa, servicios en línea y juegos de mesa para proporcionar recursos que den respuesta a esta necesidad. Ha habido muchos intentos de incluir materias de alfabetización digital en el currículo, en general tímidos y erráticos, sin estabilidad por la falta de resultados positivos. Y el balance de todo este esfuerzo ha sido casi nulo en crear habilidades de programación en la enseñanza obligatoria, que a día de hoy sólo se adquieren a nivel universitario cuando se opta por esta ingeniería.

Esto nos deja en una alfabetización computacional de la población en torno al 1%, y 9 de cada 10 son hombres. Aunque hay algo más que ha cambiado, de aquellos limitados y elitistas ordenadores personales, hemos pasado a equipos portátiles de una gran capacidad de cómputo, almacenamiento y comunicación, a una sociedad que ha adoptado sin condiciones los dispositivos móviles, smartphones, tabletas, relojes inteligentes y una amplia gama de gadgets para salud, movilidad, deporte, vigilancia y casi cualquier faceta de la vida. El Internet de las cosas integrará en un mismo protocolo a todas las máquinas domésticas o industriales, todo será accesible, programable.

Las transformaciones en el plano económico y social han sido enormes. La vida ya no se concibe sin un teléfono móvil (prueben a hacerlo), la comunicación por vía digital ha despegado con fuerza y la reciente pandemia sólo ha acelerado los cambios que se avecinaban. El comercio por Internet, la compra de bienes, los servicios, crean nuevos modelos que compiten con, y a veces sustituyen, las tiendas de barrio, el taxi o el hotel. Hasta el autostop se hace ahora en línea. Las empresas tecnológicas han desbancado a las petroleras y grandes grupos financieros en capitalización, comparable al PIB de algunos países y acumulan tanto poder como para abordar líneas de negocio en exploración espacial.

Y, en este contexto de tecnificación acelerada, somos meros consumidores, aceptamos las normas del juego sin preocuparnos de cómo funciona el móvil o la SmartTV. Se podría decir que hemos recibido las nuevas tecnologías como adoptamos el teléfono o el televisor, sin tener idea de electrónica. Pero no nos engañemos, el ejemplo no es equiparable. Un electrodoméstico cumple una única función. En cambio, el ordenador es una máquina de propósito general, de las más complejas que hemos creado. Y con conexión a Internet se puede ofrecer, virtualmente, cualquier tipo de servicio; para emprender un negocio, para solucionar problemas sociales con software libre, o para ser más competitivos en cualquier profesión, científica, ingenieril o humanística. Estos dispositivos, además, irrumpen en nuestra privacidad, haciéndonos vulnerables. La licencia para usar un móvil no se da a los 18 años, como la de conducir, ya lo utilizan desde los primeros años de vida.

Saber programar no es sólo necesario para crear software, también para entender el software que utilizamos. Y no se trata de revisar todas y cada una de las aplicaciones que instalamos en el móvil, pero si una buena parte de la población entendiera esta tecnología, alguien lo haría, y sabríamos que es mejor no instalar esta app porque contiene *malware*, o que este navegador registra todo lo que hacemos.

"Pensar computacionalmente" da una nueva forma de abordar problemas. "Programar" permite, en la práctica, resolver muchos de ellos. También da acceso a la enorme oferta laboral en el sector TIC, empleos desafiantes y bien remunerados, que en su mayoría pueden llevarse a cabo en forma remota, desde cualquier lugar del mundo. Reduciendo la brecha de género en ingeniería, ya contribuimos a reducir las brechas laboral y salarial. Enseñar programación en la enseñanza obligatoria nos prepara para una sociedad tecnificada. Sólo tiene ventajas.

En la actualidad, sin embargo, apenas hay presencia de la programación en el currículo. Necesitamos, por tanto, voluntad política para darle un lugar en la enseñanza y en las aulas. Necesitamos educadores con formación en esta área, conocedores de la terminología y con una base en programación. Y necesitamos ordenadores, que al fin es el interlocutor válido cuando se trata de aprender un lenguaje de programación.

Pues bien, para quien piense que todo esto supone un esfuerzo inabordable, tengo una buena noticia: aprender a programar, en realidad, es muy fácil. Pero antes debemos concretar qué entendemos por 'programar' y empezaré diciendo lo que no es. Programar no es hacer una aplicación móvil, no es armar un servicio web, no es diseñar robots que eviten obstáculos, ni crear sistemas embebidos para mecanismos de riego automático, tampoco es controlar drones ni detectar objetivos aéreos, no es imprimir objetos 3D. Quizás les sorprenda si afirmo que la gran mayoría de los egresados en carreras de informática no sabría hacer nada de esto. Entonces, ¿por qué pretendemos que lo hagan estudiantes de primaria?

Enseñar a programar es enseñar un lenguaje muy básico. Con una sintaxis explícita y clara, muchísimo más sencilla que la de cualquier lenguaje de comunicación humano. Con unas estructuras computacionales elementales: instrucciones, iteradores, condicionales y variables. Punto. Cuando una niña conoce estas estructuras y es capaz de combinarlas para resolver un problema, podemos afirmar que sabe programar. A partir de ahí, se puede ampliar su conocimiento llevándole a paradigmas de desarrollo de software como el diseño orientado a objetos para grandes sistemas, la programación basada en eventos para tecnología web, los lenguajes de la inteligencia artificial, la ciencia de datos y el *bigdata*, la robótica móvil, la sensorica y domótica, etc.

Es decir, como en cualquier área de conocimiento, esta casa comienza por los cimientos, no por el tejado. Porque una cosa es entender las bases de un programa, y otra muy distinta y mucho más compleja hacer que un programa tenga una funcionalidad en el mundo real, que resuelva problemas de la vida cotidiana. La formación tecnológica necesaria sólo para publicar una página web, o una app en las *appstores* es enorme y requeriría una cualificación y un tiempo fuera de las posibilidades de la educación obligatoria.

Y, si bien toda esta tecnología puede ser de gran valor para motivar, quizás nos está desviando del verdadero objetivo. Como informático y como educador, valoro mucho más que el alumnado aprenda el concepto de condicional múltiple en un entorno simplificado, que interactuar con un microcontrolador que enciende la luz cuando alguien entra en una sala. E insisto: no estoy en contra de su uso, si el estudiante ya entiende la lógica que muestrea el sensor de presencia y activa el interruptor.

Por tanto, propondré enumerar cinco elementos que considero necesarios para abordar con cierta garantía de éxito la enseñanza de la programación:

Primer elemento: un lenguaje de programación lo más próximo posible al lenguaje humano, que pueda leerse como se lee un texto, sin más abstracciones, y casi permita entender un programa, o deducir su significado, sin explicaciones previas. Esto da confianza al alumno, no es algo tan alejado de lo que conoce. Y un lenguaje que le permite cometer errores sintácticos, porque equivocándose es como mejor se induce la sintaxis.

El segundo elemento es la evidencia científica: diseñar un itinerario formativo basado en experiencias previas en el aula, con muestras representativas de alumnos, distribución geográfica y edades. Porque sabemos cómo se aprenden las matemáticas, a qué edad debe introducirse cada concepto, pero muy poco de cómo se aprende a programar (por eso estamos hoy aquí).

El tercer elemento es el ordenador mismo y el software que permite enseñar la programación. Herramientas que lleven al alumnado desde los conceptos más básicos hasta los más complejos, preferiblemente sin manuales, de forma ludificada, aprender-haciendo, con ejemplos de cómo funciona cada concepto computacional. También con información útil para seguir el progreso y estado actual de cada estudiante.

El cuarto elemento es, quizás, el más importante: profesorado acreditado para la enseñanza de la programación. No hablo de profesionales TIC, sino de maestras y profesores que han demostrado unos conocimientos mínimos en la comprensión y desarrollo de programas, y en la herramienta para su enseñanza. El buen docente logrará la motivación necesaria y el adecuado balance de esfuerzos para que no sólo los más dotados, sino la mayoría del grupo, adquieran la competencia y entiendan la belleza y utilidad de este nuevo lenguaje.

El último elemento es más sutil, pero igualmente importante: la soberanía de los datos. Si vamos a enseñar programación de forma masiva en plataformas online, los metadatos de enseñanza que almacenan deben estar ubicados en el país que los genera, de acuerdo a las leyes de privacidad de datos vigentes. No debemos olvidar que el talento es una de las materias primas más valiosas en la Economía del Dato en que estamos ya inmersos y el acceso a esta información puede ser crítico para mantener o perder el talento que nuestros propios centros están formando.

Sé que esto son sólo indicaciones generales, y en el proceso surgirán cuestiones, algunas son abordables de entrada: ¿cómo formar al profesorado? *mutatis mutandi*, las facultades de educación pueden seguir el mismo procedimiento para formar a los futuros educadores; ¿de qué herramientas dispondremos?, existen muchas opciones gratuitas, otras no gratuitas, lo importante es que sean de fácil implantación en el aula y dispongan del material complementario, como guías didácticas, solucionarios y material audiovisual complementario; o ¿dónde intervenir?, mi opinión es que se necesita intervenir poco, pero bien, una asignatura en primaria con este lenguaje elemental y otra en secundaria con un lenguaje profesional podría ser suficiente.

Por tanto, ante la pregunta planteada en este webinar, ¿cómo enseñar programación y pensamiento computacional?, mi respuesta corta sería: enseñando un lenguaje de ordenador sencillo y muy próximo al lenguaje humano. La respuesta larga aclararía que el pensamiento computacional ya se adquiere programando: reconocimiento de patrones, abstracción, modularización son habilidades básicas sin las cuales es imposible programar; añadirlas transversalmente en la formación pedagógica, me parece una excelente idea, todo suma. También incidiría en que es la vía más rápida, efectiva y práctica: mejoraremos la empleabilidad de nuestros jóvenes, les haremos más competentes en cualquier profesión que desarrollen, y, seamos honestos, esto para un país tiene un gran valor.

Por último, quiero incidir en un aspecto menos visible, pero igualmente importante: no se puede privar a nuestras niñas, niños y jóvenes del placer de expresarse en estos nuevos lenguajes, de construir y crear soluciones, no sólo para el disfrute, también para hacer frente al mundo complejísimo que dejamos en sus manos.

Todo esto es posible, si entendemos que enseñar a programar puede ser tan sencillo como enseñar a manejar un auto.