

Estado de la enseñanza de la programación en Costa Rica

Andrés Navas, Pablo Pérez, and Federico Jiménez

Escuela de Ingeniería,
Universidad Latinoamericana de Ciencia y Tecnología,
ULACIT, Urbanización Tournón, 10235-1000
San José, Costa Rica

anavasa878@ulacit.ed.cr, jperez027@ulacit.ed.cr, fjimenezg001@ulacit.ed.cr
<http://www.ulacit.ac.cr>

Abstract. The increasing use of technology by younger generations as part of their daily activities, such as interacting in social networks and playing video games on mobile handheld devices, is reflecting in the efforts to introduce computer programming in education curricula of countries with high education levels.

Costa Rica, ranked 6th by human development index in the Latin-American region by the United Nations in 2014 is no stranger to similar initiatives, thanks to the Ministry of Public Education and the Omar Dengo Foundation in their joint undertaking: PRONIE.

By analyzing the contents of this endeavor and comparing them to acclaimed international equivalents, this work seeks to find Costa Rica's status regarding coding in primary and secondary school classrooms as well as to identify opportunities for future improvement.

Keywords: education, computer programming, teaching, coding, children, Costa Rica

1 Introducción

Los avances en la tecnología y la informática (en especial las tecnologías móvil y web) influyen de forma particular a los niños y adolescentes. En general, se ha creado un mercado que requiere de un mayor nivel de conocimiento técnico sobre la tecnología y el desarrollo de sistemas por parte de los usuarios. Esto incide en una mayor demanda por parte de las empresas, de personas con conocimientos necesarios (Jang, 1992).

La enseñanza de la programación durante la educación primaria y secundaria estimula desde temprana edad la capacidad para crear aplicaciones. Un país con estudiantes mejor preparados tiene mayores ventajas; brinda beneficios como el ofrecer futuras oportunidades para el crecimiento social y laboral de los estudiantes. Al mismo tiempo, el país se beneficia al incrementar dicha capacitación. Inclusive puede beneficiar al país, con personas jóvenes mejor capacitadas para impulsar con ideas innovadoras, el rápido desarrollo de la tecnología (Brusilovsky & Others, 1994).

Existen iniciativas en algunos países para promover que los niños y adolescentes aprendan a utilizar y desarrollar aplicaciones de baja y mediana complejidad, para computadoras, dispositivos móviles e Internet. Por ejemplo, *Hour of Code* es una de estas iniciativas apoyada por Bill Gates y Mark Zuckerberg en el 2013. Ésta es una organización sin fines de lucro, y su objetivo es habilitar la enseñanza de la programación en más escuelas y colegios. (“Code.org”, 2014a).

En Costa Rica, los planes de educación informática en la primaria y la secundaria incluyen el aprendizaje de herramientas como *MicroMundos*, *Scratch* y *Alice* para el desarrollo de aplicaciones sencillas. (“Unicef”, 2014).

También hay programas para la enseñanza de desarrollo de *software* para estudiantes de décimo, undécimo y duodécimo año. En esos programas se define las directrices para la enseñanza de la programación en colegios técnicos; además se determina las herramientas y lenguajes de programación que se enseñará a los estudiantes (“MEP”, 2009).

En este contexto, esta investigación lleva a cabo un análisis de los esfuerzos que se realiza a nivel internacional y nacional, con el fin de promover la programación en la educación primaria y secundaria. Por consiguiente, este trabajo busca comparar el enfoque de programas educacionales, proponer mejoras al plan de estudio actual y establecer si es necesario cambiar el perfil del educador.

2 Antecedentes

Previo a la década de los ochenta y debido a los altos costos de los equipos tecnológicos, la enseñanza de la computación y por ende, la programación, se impartía prácticamente solo en universidades; es decir, aproximadamente a partir de los 18 años.

A partir de la revolución de la PC y gracias a la reducción de costo de equipos, el acceso a computadores se generalizó, extendiéndose así su enseñanza a otras instituciones aparte de las universidades.

En años recientes, se han popularizado distintas iniciativas que promueven la enseñanza de la programación a niños desde más temprana edad. Éstas no pretenden convertir a todos los niños en programadores, sino desarrollar su capacidad de abstracción, análisis y razonamiento lógico como herramientas para resolver problemas. Se cree que entre más temprana la edad de estimulación, mayor el beneficio a su aprendizaje de las artes, las ciencias, etc. (“Computing at School Working Group”, 2012)

En algunos países, las iniciativas de enseñanza de la programación son impartidas por organizaciones sin fines de lucro, mientras que en otros casos los programas son apoyados por el gobierno.

En Estados Unidos, la Fundación Bill & Melinda Gates es patrocinadora de la iniciativa Hour of Code (“Code.org”, 2014a), con el fin de enseñar programación a los niños de escuelas, en el estado de Washington.

En Reino Unido, se creó la fundación Codeclub.org que fomenta la educación temprana de programación en jóvenes estudiantes en varias escuelas locales. Este

club promueve planes de capacitación para profesores y ayuda a crear clubes en otros países (“CodeClub.org”, 2014).

En el año 2012, Estonia implementó un programa llamado *ProgeTiiger* para enseñar a programar a niños, desde los 7 a los 19 años (“ProgeTiiger.ee”, 2014).

En Costa Rica, la Fundación Omar Dengo junto con el Ministerio de Educación Pública, han desarrollado diferentes versiones del Programa Nacional de Informática Educativa (PRONIE) desde 1988. El programa, con la colaboración entre ambas instituciones, genera los siguientes beneficios: acceso a los recursos disponibles en el PRONIE (laboratorios de Informática Educativa para el aprendizaje de la programación y materiales educativos), impulsa la utilización de tecnologías de información y comunicación dentro de los procesos académicos y fomenta el desarrollo de nuevas e innovadoras tecnologías. Actualmente las escuelas y colegios aplican la versión más reciente de este programa, del 2010. (“Fundacion Omar Dengo”, 2013).

El IV Informe del Estado de la Educación menciona que una de las aspiraciones de la educación en Costa Rica es permitirle a los niños y adolescentes aprovechar los avances de la ciencia y la tecnología, con el fin de mejorar su calidad de vida y desarrollarse como seres humanos capaces de aportar a la sociedad (“Programa Estado de la Nación”, 2013).

Sin embargo, este mismo informe revela que el PRONIE no ha logrado sobrepasar el 65% de cobertura a nivel nacional, tanto en instituciones de educación primaria, como secundaria. otros elementos a considerar son:

1. Existen claras diferencias de cobertura de este programa entre la gran área metropolitana y las zonas rurales.
2. La efectividad del programa es mayor entre los colegios científicos, técnicos y bilingües, y es menor entre los colegios nocturnos y aquellos que se encuentran ubicados en zonas con poco apoyo institucional.

Un aspecto notorio del primer informe sobre el Estado de la Ciencia, la Tecnología y la Innovación (“Programa Estado de la Nación”, 2014) es que no hace mención de las competencias o el estado de la educación primaria y secundaria, con relación a la tecnología y la innovación.

A pesar de que sí se hace mención al desafío de atraer más jóvenes hacia las áreas de ciencia y tecnología a nivel universitario, no se toma en consideración la programación o informática, sino que únicamente las ciencias y la matemática, y las competencias de los jóvenes a desarrollar para ese propósito.

3 Desarrollo

Para poder comparar la enseñanza de la programación en Costa Rica, se mencionó un conjunto de iniciativas de países líderes en educación (“Pearson”, 2014), así como de otros que han sido reconocidos por sus esfuerzos recientes (“OECD.org”, 2014).

En la tabla 1 se muestra los programas de cada país investigado para realizar la comparación.

Programa	País
Hour of Code	Estados Unidos
Code Club	Reino Unido
ProgeTiiger	Estonia
PRONIE	Costa Rica

Table 1. Programas por País

3.1 Estados Unidos

Hour of Code es un programa de la organización Code.org, el cual ofrece cursos en dos tipos de modalidades: en la primera, los cursos tienen requisitos mientras que en la segunda no.

En la primera modalidad, el programa *Computer Science for Elementary School de Hour of Code* (“Code.org”, 2014b) está compuesto de cuatro cursos. Cada curso enseña distintos conceptos de programación dirigidos a niños de distintas edades y como requisito se debe completar el curso predecesor, como se muestra en la tabla 2.

	Edades	Contenidos
Curso 1	4 a 6	Secuencias Depuración Ciclos
Curso 2	6 a 8	Algoritmos Secuencias Ciclos Programación Depuración Condicionales Binario
Curso 3	8 a 10	Funciones Condicionales Ciclos Depuración Internet Ciudadanía Digital
Curso 4	10	Algoritmos Ciclos Variables Parámetros Binario

Table 2. Hour of Code, Estados Unidos

En la segunda modalidad, los cursos de *Hour of Code* se presentan como juegos en línea, interactivos y de acceso libre, sin restricción por edad. Los mismos

permiten que cada estudiante aprenda de manera individual o en grupo sobre lenguajes de programación específicos (“Code.org”, 2014c), tales como:

- Java
- JavaScript
- Desarrollo de videojuegos
- Desarrollo de aplicaciones móviles
- Python
- Swift (lenguaje de programación)
- Quórum (lenguaje de programación)
- App Inventor
- Binario

Debido a que los programas de estudio de la programación son optativos y a menudo no se toman en cuenta en las calificaciones escolares, *Hour of Code* sugiere que en Estados Unidos se defina las bases de cursos obligatorios en la enseñanza de Ciencias de la Computación. Adicionalmente, se debe establecer el perfil de los profesores y el mecanismo de certificación para demostrar que están preparados para enseñar este tipo de cursos. (“Code.org”, 2013)

3.2 Reino Unido

En el Reino Unido se identificó proyectos de enseñanza de programación a niños, uno público y otro privado. Codeclub.org es una organización sin fines de lucro que ofrece un programa con cursos gratuitos para estudiantes entre los 9 y los 11 años de edad, así como capacitaciones para profesores, con un costo asociado (“CodeClub.org”, 2014).

CodeClub.org tiene tres niveles de enseñanza: en el primer nivel se enseña al estudiante a crear un juego de video de forma divertida y creativa, utilizando la herramienta *Scratch*. El segundo nivel cubre la construcción de sitios y páginas web mediante HTML y CSS, mientras que en el último nivel se aprende el lenguaje Python. Los cursos no son evaluados con calificaciones, sino que consisten en guías con tareas por completar. En la tabla 3 se muestran los contenidos de cada nivel.

	Edades	Contenido	Herramientas
Nivel 1 Modulo 1	9 a 11	Secuencias Sprites Loops (finitos e infinitos) Variables, Listas Funciones, Condicionales Algoritmos	Scratch
Nivel 2 Modulo 1	9 a 11	Codigo Fuente HTML Listas, Enlaces, Títulos, Imágenes Formato y Estilo Introduccion a Hojas de Estilo CSS	Scratch Editor Texto HTML CSS
Nivel 2 Modulo 2	9 a 11	Layout HTML y CSS Diseño y Estilo CSS avanzado Audio y Video	Scratch Editor Texto HTML CSS Presentaciones
Nivel 3 Modulo 1	9 a 11	Sintaxis Errores Lógicos Hileras de caracteres Tipos de datos Casting de tipos Variables, Condicionales, Ciclos Random, Listas y Arreglos	Python 3.2 Editor de texto
Nivel 3 Modulo 2	9 a 11	Cifrado y Encriptamiento Juegos AI Gráficos 2D	Python 3.2 Editor de texto

Table 3. Code Club, Reino Unido

Por otra parte, el Ministerio de Educación de Reino Unido definió en el año 2013 un programa actualizado sobre la enseñanza de la computación (“GOV.UK”, 2013a). En septiembre del 2014, el programa empezó a operar en todas las escuelas públicas. El programa se organiza en cuatro etapas según rangos de edades: de 5 a 7 años, de 7 a 11 años, de 11 a 14 y de 14 a 16 años, cuyos contenidos se muestran en la tabla 4:

	Edades	Contenidos
Etapa 1	5 a 7	Algoritmos Crear y depurar programas simples Razonamiento lógico Crear, organizar y consultar medios digitales Reconocer usos comunes de la tecnología fuera de la escuela Privacidad de la información Dónde consultar o solicitar ayuda
Etapa 2	7 a 11	Escribir y depurar programas con objetivos específicos Controlar y simular problemas físicos Descomposición de problemas grandes en más pequeños Ciclos y variables Redes de computadoras Uso selectivo del contenido digital Recolectar, evaluar, analizar, presentar datos e información Uso de tecnología segura y responsable
Etapa 3	11 a 14	Abstracción y diseño de problemas Algoritmos computacionales como búsqueda y ordenamiento Listas, arreglos y otras estructuras de datos Procedimientos y funciones Hardware y Software Uso de tecnología segura y responsable (identidad y privacidad)
Etapa 4	14 a 16	Desarrollo de capacidades de creatividad en ciencias de la computación Análisis de resolución de problemas mediante procedimientos computacionales Identificar cómo la tecnología expone privacidad e identidad Reportar dudas y a quien acudir.

Table 4. Programa Nacional Reino Unido

3.3 Estonia

La noticia de un ambicioso programa educativo que llevaría la enseñanza de la programación al 100% de las escuelas primarias de Estonia se dio a conocer en el 2012. Se trataba de la iniciativa *ProgeTiiger* del Centro de Innovación de la Fundación para la Educación de Tecnologías de Información de Estonia (Olson, 2012).

ProgeTiiger, que en estoniano significa tigre de la programación, ya ha sido implementado en escuelas estonianas. En la Tabla 5 se muestran los contenidos de su programa.

	Edades	Contenidos	Herramientas
Introducción	7 a 10	Utilización de la computadora Conceptos de lógica Programación básica Secuencias, Bucles Condiciones, Variables Abstracción (Subrutinas) Recursividad	Robozzle Kodu Game Lab LOGO
Programación de Juegos	11 a 15	Creación de Videojuegos Aplicación de conceptos de Programación Secuencias, Bucles, Condiciones, Variables Concurrencia, Sincronización Lógica Booleana Manejo de Eventos Diseño de Interfaz Estructuras de datos Procedimientos y funciones	Scratch AppInventor Python
Robótica	15 a 17	Programación para micro-computadoras Controladores digitales	LEGO WeDo LEGO NXT-G LEGO NXC
Desarrollo Web	14 a 16	Creación y diseño de páginas web Código HTML Hojas de Estilo Aplicaciones web cliente-servidor	HTML, CSS JavaScript PHP MySQL

Table 5. ProgeTiiger, Estonia

A la fecha, se han realizado toda una serie de concursos a nivel nacional en programación y robótica, en los que se invita a estudiantes de 5 a 19 años a participar (Roonema, 2014).

Al ser reconocido como uno de los países mejor interconectados en Europa, el Ministerio de Ciencia y Educación de Estonia trabaja en la creación de una nube educativa en conjunto con su contraparte gubernamental finlandesa, en la que sistemas de información estarán disponibles de manera pública para compartir los recursos didácticos de ambas naciones. La fase de pruebas de dichos sistemas iniciará en Mayo del 2015. (OECD.org, 2013).

3.4 Costa Rica

En Costa Rica, los Laboratorios para la enseñanza de la informática forman parte del PRONIE (Programa Nacional de Informática Educativa), establecido por el Ministerio de Educación Pública y la Fundación Omar Dengo. Los laboratorios están dirigidos a estudiantes de Preescolar, I, II, y III Ciclo en colegios públicos (MEP-FOD, 2011), como se muestra en la Tabla 6:

	Edades	Contenido	Herramientas
Preescolar	4 a 6	Uso y cuidado de la computadora Funcionamiento de las redes Animación digital	Micromundos
I Ciclo	7 a 9	Resolución de Problemas Razonamiento Lógico Razonamiento matemático Eventos y Objetos Carpetas y Archivos Introducción a Scratch Código y Editores de texto Solución de Problemas Creación de un Juego Búsquedas de Internet Blogs e Internet	Micromundos Power Point Ispring Scratch
II Ciclo	10 a 12	Periódico Digital Conceptos de Programación Creación de un Juego Condicionales Operadores Lógicos Variables	Scratch
III Ciclo	13 a 15	Algoritmos Pseudocódigo Estructuras de datos Diagramas de flujo Programación	

Table 6. I, II, III Ciclo Costa Rica

En Costa Rica, los programas establecidos para la enseñanza de lenguajes de programación son específicos para colegios técnicos y están diseñados para estudiantes del décimo, undécimo y duodécimo años, es decir, del ciclo de educación diversificada (MEP, 2009).

Además, existe el programa llamado Talleres Exploratorios, el cual incluye diversos aspectos de tecnologías de información y comunicación el cual se imparte en tres niveles: séptimo, octavo y noveno año. Es en este último año en el que se imparte enseñanza de la programación (MEP, 2006) según se muestra en la Tabla 7.

	Edades	Contenido	Herramientas
Noveno	15	Algoritmos Funciones Plantamiento de Problemas Análisis y Diseño de Solución Pruebas Validaciones Representación de Algoritmos Solución de Problemas Diagramas de Flujo Método Axiomático Cálculo Proposicional Lenguaje Simbólico Condicionales, Decisión, Ciclos Variables, Procedimientos, Funciones	Micromundos

Table 7. Taller Exploratorio

Los programas actuales son utilizados como un medio interactivo para el aprendizaje de temas como los recursos naturales, las leyes de tránsito, matemática, ciencias, entre otros, enfocándose más en la capacitación en el uso de la herramienta específica (p. ej. MicroMundos) que en teoría de programación. La excepción a lo anterior se da en el noveno grado, en el cual se enseña sobre diagramas de flujo y se desarrolla un videojuego empleando el entorno de programación en tres dimensiones llamado *Alice*, e incluso se presenta un taller exploratorio de tecnologías de la información y comunicación con una especialidad en Informática en programación, y como se mencionó anteriormente, estos talleres brindan capacitaciones sobre la programación (MEP-FOD, 2011).

	Edades	Contenidos
Décimo	16	Generalidades de la Computación Uso de Paquetes de Software Diseño Web Sistemas Especializados de Información Conectividad
Undécimo	17	Programación Introducción a Estructuras de Datos Implementar Estructuras (arreglos, árboles, punteros, registros, y cadenas) Programación Orientada a Objetos Principios de Calidad en Software Mercadeo Administración de Proyectos Informáticos
Duodécimo	18	Aplicar conceptos: Funciones y Herramientas (lenguaje de programación específico) Desarrollo Web (lenguaje de programación específico) Desarrollo de Aplicaciones de Escritorio (.NET)

Table 8. Educación Diversificada

4 Discusión

Los programas de enseñanza de la programación que fueron investigados contienen similitudes entre sí; por ejemplo, los temas principales y las edades en que son impartidos, así como los entornos y herramientas utilizadas.

4.1 Herramientas

La tabla 9 muestra la comparativa realizada por programa y las herramientas utilizadas. Elementos que se desprenden de esta tabla:

	USA Hour of Code	UK Code Club	EE ProgeTiiger	CR PRONIE
LOGO			X	X
Scratch	X	X	X	X
AppInventor	X		X	
Alice				X
HTML/CSS		X	X	X
Python	X	X		
Visual Basic				X
JAVA	X	X		
JavaScript	X	X	X	
Lego WeDo Lego NXT-G LEGO NXC			X	
PHP MySQL			X	

Table 9. Herramientas

1. Estonia enseña la programación incluyendo la mayor cantidad de herramientas y lenguajes de programación.
2. La herramienta *Scratch* es utilizada en los 4 programas evaluados.
3. Costa Rica utiliza las herramientas *Scratch*, *Alice* y *AppInventor*, también utilizadas en los otros países investigados.
4. Solamente Costa Rica enseña sobre la herramienta *Visual Basic*.

4.2 Edad Mínima

Se realizó una comparación de los principales temas en los contenidos de los programas investigados. En la siguiente tabla, se muestra la comparación en la cual se refleja la edad mínima a la que se expone a los alumnos los temas más relevantes en la enseñanza de la programación.

Contenidos	USA Hour of code	UK Code Club	EE Progetiiger	CR PRONIE
Razonamiento Lógico	4	7	7	9
Conceptos Básicos de Programación	6	7	7	9
Depuración de código	4	5	7	9
Conceptos avanzados de programación	6	11	11	15
Robótica	-	-	15	-
Desarrollo web	9	9	17	16

Table 10. Contenidos por Edades

El introducir desde edades tempranas el razonamiento lógico y la capacidad de analizar y abstraer problemas beneficia a los estudiantes. Puesto que la

computación se relaciona con matemáticas y otras ciencias, la misma puede ser utilizada como método para la resolución de problemas aplicando conocimientos de programación y funcionamiento de sistemas digitales (“GOV.UK”, 2013b).

Del cuadro comparativo se obtienen las siguientes conclusiones:

1. Los programas investigados introducen los conceptos más relevantes de la programación a diferentes edades en cada país. En el caso de Estados Unidos, el programa *Hour of Code* expone a los estudiantes a estos temas a más temprana edad. Por ejemplo, el razonamiento lógico y la depuración de código son incluidos en el programa para niños a partir de los 4 años, mientras que en el caso de Costa Rica no se incluye este contenido sino hasta los 9 años.
2. En contenidos de conceptos básicos de programación, los demás países introducen los temas a los estudiantes desde los 6 y 7 años, mientras que en Costa Rica hasta los 9 años, lo cual evidencia un rezago en comparación con las iniciativas internacionales.
3. Los conceptos avanzados de programación presentan mayores diferencias con respecto a otros en la edad mínima de estudio en cada país. En Estados Unidos, *Hour of Code* los introduce desde los 6 años, en Reino Unido y Estonia desde los 11 años, mientras que en Costa Rica hasta los 15 años.
4. El Desarrollo Web se introduce en Costa Rica hasta los 16 años, mientras que en Estados Unidos y Reino Unido, los mismos temas se desarrollan inicialmente a partir de los 9 años. Destaca el caso particular de Estonia, pues en este país se incluye otra temática en edades inferiores y el Desarrollo Web se imparte a partir de los 17 años.
5. La robótica es el tema que se incluye en los currículos antes del desarrollo web en Estonia.
6. Solamente en Estonia se incluye robótica en el programa nacional.
7. En Costa Rica, la robótica se enseña en un curso opcional de la Fundación Omar Dengo, mismo que tiene costo y no forma parte del programa educativo para todos los estudiantes.

5 Conclusiones

Los programas de Informática Educativa en Costa Rica para escuelas primarias contienen temática similar y aplican las mismas herramientas que los planes de enseñanza de la programación en los otros países investigados.

Según las guías didácticas del PRONIE, la herramienta *Scratch* se utiliza en diferentes años lectivos. Puesto que este software requiere conexión permanente a Internet, y en Costa Rica al menos un 55% de las escuelas públicas tiene conexión a Internet de baja velocidad (“Nacion.com”, 2014), lo cual entorpece el uso de la herramienta en las instituciones donde se presenta esta desventaja.

Aparte de los obstáculos de conectividad en las escuelas, se encontró un rezago en la edad inicial en la que se introduce conceptos de programación, tanto básicos como avanzados, con respecto al resto de programas.

En Costa Rica se introduce los temas básicos a los 9 años, mientras que los temas avanzados aparecen hasta los 15 años. Según la comparativa realizada en

la sección anterior, el rezago ante Estonia y Reino Unido para la introducción de conceptos básicos de programación es de 2 años; para conceptos avanzados, el rezago es de 4 años. Frente a Estados Unidos y el programa Hour of Code, el retraso es de 5 y 9 años respectivamente.

Mientras que en Estonia y Reino Unido los planes de enseñanza de programación son obligatorios en todas las escuelas públicas, en Costa Rica, los cursos de Informática Educativa del PRONIE son aplicados como talleres opcionales en los colegios públicos, por tanto, la calificación del estudiante no se considera para el promedio de notas finales del año lectivo. En contraposición, en los colegios técnicos nacionales, la calificación es incluida para el promedio anual, como la de cualquier otro curso fundamental que es requisito para aprobar el año, tal como en los países mencionados anteriormente.

Las iniciativas internacionales ofrecen cursos opcionales que preparan a los estudiantes para incursionar en ámbitos profesionales desde edades tempranas. En Estonia, por ejemplo, el plan nacional de educación ofrece cursos de robótica, permitiendo a los estudiantes aprender de disciplinas ingenieriles. Asimismo, *Hour of Code* en Estados Unidos, incluye un curso para aprender sobre el emprendedurismo.

En contraste, la Fundación Omar Dengo ofrece solamente un curso de robótica con costo para grupos pequeños dentro de las instalaciones de dicha institución y la formación profesional aparece hasta a partir del noveno año con los Talleres Exploratorios. No obstante, estos cursos no se enfocan en la resolución de problemas utilizando herramientas de forma creativa e innovadora, sino en el uso básico de las mismas.

Tanto PRONIE como el resto de programas analizados, ofrecen recursos en línea para la capacitación de docentes. Para lograr el mayor aprovechamiento y enriquecer su calidad como profesores, es crucial optimizar la utilización de estos programas de educación al educador.

6 Recomendaciones

Basados en los datos encontrados en la investigación y comparación de los programas de enseñanza de la programación en los países evaluados, se identifican las siguientes oportunidades de mejora:

1. Auditar el cumplimiento de las guías didácticas del PRONIE para todas las escuelas y colegios participantes del programa.
2. Incluir la calificación de los cursos de informática educativa en el promedio final del año escolar.
3. Impartir a más temprana edad los cursos de informática educativa que incluyen temas de razonamiento lógico y conceptos de programación.
4. Incorporar al programa de Laboratorios de Informática Educativa la enseñanza de lenguajes de programación, tales como Java y Python.
5. Calibrar el nivel de capacitación de los profesores de informática educativa para homogenizar la manera en que se imparten los cursos del PRONIE.

References

- Brusilovsky, P., & Others. (1994). *Teaching Programming to Novices A Review of Approaches and Tools*. Retrieved from <http://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=ED388228&lang=es&site=ehost-live> pages 1
- Codeclub.org. (2014). *About Code Club*. Retrieved from <https://www.codeclub.org.uk/about> pages 3, 5
- Code.org. (2013). *Convince your School*. Retrieved from http://code.org/files/convince_your_school_or_state.pdf pages 5
- Code.org. (2014a). *Promote Computer Science*. Retrieved from <http://code.org/promote> pages 2
- Code.org. (2014b). *Computer science for elementary school*. Retrieved from <http://code.org/educate/k5> pages 4
- Code.org. (2014c). *Learn an Hour of Code*. Retrieved from <http://code.org/learn> pages 5
- Computing at school working group. (2012). *Computer Science as a School Subject: Seizing the Opportunity*. Retrieved from <http://academy.bcs.org/sites/academy.bcs.org/files/cs-school-subject.pdf> pages 2
- Fundacion omar deno. (2013). *Cobertura Nacional Centros Educativos*. Retrieved from http://www.fod.ac.cr/index.php?option=com_content&view=article&id=77&Itemid=185 pages 3
- Gov.uk. (2013a). *National curriculum in England: computing programmes of study*. Retrieved from <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study> pages 6
- Gov.uk. (2013b). *National curriculum in England: computing programmes of study*. Retrieved from <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study> pages 13
- Jang, Y. (1992). Cognitive transfer of computer programming skills and analogous problem solving. *National Meeting of the American Educational Research Asociation*. pages 1
- MEP. (2006). Mep.go.cr. *Taller Exploratorio Herramientas para la Programación de Computadoras*. Retrieved from <http://www.mep.go.cr/sites/default/files/descargas/programas-de-estudio/tallerherramientasprogra9.pdf> pages 9
- Mep. (2009). *Programas de Estudio*. Retrieved from [http://www.mep.go.cr/programa-estudio?page=1&keys=Informatica&term_node_tid_depth\[hsid\]=3&term_node_tid_depth\[hierarchical_select\]\[selects\]\[0\]=Any&term_node_tid_depth\[flat_select\]=Any&hs_form_build_id=hs_form.6a24ecda00bbc04dde25f2de842017a2](http://www.mep.go.cr/programa-estudio?page=1&keys=Informatica&term_node_tid_depth[hsid]=3&term_node_tid_depth[hierarchical_select][selects][0]=Any&term_node_tid_depth[flat_select]=Any&hs_form_build_id=hs_form.6a24ecda00bbc04dde25f2de842017a2) pages 2
- MEP. (2009). Mep.go.cr. *Programa de Estudio Educación Diversificada Técnica*. Retrieved from <http://www.mep.go.cr/sites/default/>

- files/descargas/programas-de-estudio/infoprogramacion12.pdf
pages 9
- MEP-FOD. (2011). Pronie mep-fod. *Guías Didácticas*. Retrieved from <http://www.fod.ac.cr/guiasdidacticas/> pages 8, 10
- Nacion.com. (2014). *Baja velocidad de Internet limita uso en 55% de escuelas*. Retrieved from http://www.nacion.com/nacional/educacion/Mayoria-escuelas-colegios-Internet-velocidad_0_1430456994.html pages 13
- OECD.org. (2013). Oecd.org. *Education at Glance 2013*. Retrieved from <http://thelearningcurve.pearson.com/> pages 8
- Oecd.org. (2014). *PISA 2012 Results*. Retrieved from <http://www.oecd.org/pisa/keyfindings/pisa-2012-results-overview.pdf> pages 3
- Olson, P. (2012). Forbes. *Why Estonia Has Started Teaching Its First-Graders To Code*. Retrieved from <http://www.forbes.com/sites/parmyolson/2012/09/06/why-estonia-has-started-teaching-its-first-graders-to-code/> pages 7
- Pearson. (2014). *The Learning Curve*. Retrieved from http://thelearningcurve.pearson.com/content/download/bankname/components/filename/The_Learning_Curve_2014-Final_1.pdf pages 3
- Progetiiger.ee. (2014). *Õppematerjalid*. Retrieved from <http://progetiiger.ee/%C3%B5ppematerjalid> pages 3
- Programa estado de la nación. (2013). *Cuarto Informe del Estado de la Educación*. Retrieved from http://www.estadonacion.or.cr/files/biblioteca_virtual/educacion/004/castro_desempeno-ed-basica-y-diversificado.pdf pages 3
- Programa estado de la nación. (2014). *Estado de la Ciencia la Tecnología y la Innovación*. Retrieved from <http://estadonacion.or.cr/estado/Publicaciones/estado-ciencia-tecnologia-innovacion-1/index.html> pages 3
- Roonema, H. (2014). Progetiger, lego robots and computer-based math conquer schools. *Life in Estonia*. Retrieved from http://issuu.com/eas-estonia/docs/lie_spring_2014_issuu/49 pages 8
- Unicef. (2014). *Programa TIC y Educación Básica*. Retrieved from <http://www.unicef.org/argentina/spanish/Costa.Rica.WEB.pdf> pages 2