

DESARROLLAR HABILIDADES DE PENSAMIENTO COMPUTACIONAL EN ESTUDIANTES DE SECUNDARIA

Jocelyn Simmonds, Carlos Estay y Constanza Díaz



 **TECHNOVATION**
Girls Chile



MOTOROLA SOLUTIONS
FOUNDATION

ÍNDICE

INTRODUCCIÓN.....	4
CARACTERIZACIÓN DE LOS ESTABLECIMIENTOS EDUCATIVOS.....	6
TRABAJO RELACIONADO.....	8
EXPERIENCIA.....	10
DISCUSIÓN.....	15
CONCLUSIONES.....	17
REFERENCIAS.....	18

INTRODUCCIÓN

A nivel mundial, ha habido un creciente apoyo a los programas que animan a los estudiantes a codificar, como una forma de desarrollar su pensamiento computacional y sus habilidades del siglo XXI (ver p. Ej., [1, 4, 8-13, 16, 17, 19, 21, 23, 24, 26]). Evaluar estos programas es un desafío, ya que los estudiantes no sólo están expuestos a nuevos conceptos, sino que también trabajan en el desarrollo de nuevas habilidades y destrezas. También es demasiado pronto para analizar los efectos a largo plazo que estos programas tienen en los estudiantes.

El programa Technovation Girls (TG), presente en más de 110 países, es un programa donde su plan de estudios no se centra únicamente en la codificación, sino que también incluye temas de informática, espíritu emprendedor, resolución de problemáticas sociales y liderazgo empresarial. Organizado generalmente como un club extracurricular, las estudiantes trabajan durante 12 a 32 semanas en equipos de entre 2 a 5 niñas de entre 10 a 18 años para idear y crear un prototipo de una aplicación móvil que aborde uno de los Objetivos de Desarrollo Sostenible de la ONU. Los equipos tienen la posibilidad de presentar sus prototipos a jueces nacionales e internacionales, y los equipos ganadores locales avanzan al Technovation Challenge, una gran final que se celebra una vez al año. Según su sitio web, “los estudiantes expresan un mayor interés en la tecnología y el liderazgo después de participar en nuestro programa, y el 58% de nuestras egresadas se inscriben en más cursos de informática después de Technovation”.

El capítulo que se estudia en este artículo ha estado activo durante los últimos 6 años en Chile, ubicada en Sudamérica. Este capítulo ha estado participando del programa TG en su contexto local y a fines de 2019, decidió intentar ejecutar el programa TG durante el horario escolar, en el marco de la asignatura de Tecnología, aprovechando la similitudes conceptuales del currículo de la asignatura de Tecnología, se aplica metodología aprendizaje basado en proyectos y el programa de TG todo esto en lugar de después de la escuela. Esto requirió varias adaptaciones tales como: extender la duración del programa al año escolar, expandir el plan de estudios para apoyar mejor a los estudiantes de las escuelas de oficios o técnico profesionales y permitir equipos mixtos (que no participaran del internacional), la mayoría de las escuelas aquí son mixtas.

Algo importante de comentar sobre la instalación del programa, fue que el factor pandemia de Covid-19 que golpeó a principios de 2020, instancia donde las escuelas tuvieron que cambiar rápidamente a clases en línea. Cada escuela tiene al menos un maestro que trabaja con sus estudiantes en el programa TG durante el año, y cuentan con el apoyo de un profesional STEM pagado por el capítulo local de TG denominado KAM.

Dado que uno de los productos principales del programa TG es el prototipo de la aplicación móvil, se quiso estudiar si los cambios realizados en el programa TG tenían un impacto en las habilidades de pensamiento computacional de los estudiantes. Como tal, se planteó la siguiente pregunta de investigación: ¿Participar en el programa TG tiene un impacto en el desarrollo de pensamiento computacional de los estudiantes? Para responder a esta pregunta, se estudió la experiencia en 3 escuelas secundarias que ofrecen formación profesional, con características variables: localizadas en diferentes provincias de Chile. Estas tres escuelas de oficios ofrecen opciones profesionales relacionadas con STEM. Los estudiantes que participaron en el estudio estaban todos en el primero medio durante 2020, pero con edades comprendidas entre los 13 y los 18 años.



CARACTERIZACIÓN DE LOS ESTABLECIMIENTOS EDUCATIVOS

Instituto Superior de Comercio Francisco Araya Bennett de Valparaíso (INSUCO)

Establecimiento ubicado en la ciudad de Valparaíso, Región de Valparaíso. Institución mixta de formación técnica profesional que cuenta con las especialidades de Administración mención logística, Administración mención recursos humanos, Contabilidad, Programación y Servicios de turismo. Sus estudiantes tienen un índice de vulnerabilidad del 94%. El programa Technovation Girls Chile se implementó en modalidad “Capacitado y acompañado”.

Liceo Comercial de Desarrollo Temuco (LCD)

Establecimiento ubicado en la ciudad de Temuco, Región de La Araucanía. Es una institución mixta de formación técnica profesional en la que sus estudiantes se pueden especializar en Administración, Contabilidad, Gráfica y Programación. Además, cuentan con un programa de especialización dual en el que los y las estudiantes pueden formarse en el liceo y en una empresa simultáneamente. Sus estudiantes tienen un índice de vulnerabilidad de 90%. El programa Technovation Girls Chile también se implementó en modalidad “Capacitado y acompañado”.

Liceo Comercial Vate Vicente Huidobro (LCVH)

Establecimiento ubicado en la comuna de San Ramón, Región Metropolitana. Es una institución mixta que ofrece educación técnica profesional en el área técnica y comercial. Los y las estudiantes pueden especializarse en Administración mención Recursos Humanos, Administración mención Logística, Contabilidad y Programación en computación. El índice de vulnerabilidad de sus estudiantes es de 92,16%. El programa Technovation Girls Chile se implementó en modalidad “Capacitado y acompañado”, es decir, se comenzó con una capacitación previa de los profesores y profesoras de Tecnología y durante el año, se les acompañó y apoyó en la implementación del programa.



En el año 2020, el proceso de toma de los datos se realizó en dos instancias, una de entrada en marzo al inicio del año escolar y otra de salida en noviembre.

Para evaluar las habilidades de pensamiento computacional de los estudiantes, se usó la prueba desarrollada por Román-González et al. [18], dado que ya ha sido validado con una gran población de hispanohablantes en un rango de edad similar a nuestra población. Los estudiantes respondieron la prueba al comienzo del programa TG y luego nuevamente al final. Se encontró que, en general, hay una leve mejora positiva en los puntajes después de terminar el programa TG, pero con diferentes tendencias por escuela, sexo y edad de los participantes. También se vio una alta confianza de los estudiantes en sus conocimientos después del programa, aunque existen diferencias moderadas a grandes entre los puntajes esperados y reales de las pruebas. Se finaliza con una discusión sobre las diferencias en las escuelas y cómo se implementó el programa TG, lo que puede explicar estas diferencias.

TRABAJO RELACIONADO

Ha habido varias propuestas para evaluar el desarrollo de pensamiento computacional en niños en edad escolar. Aunque el pensamiento computacional se considera un proceso de pensamiento cognitivo, varios autores se centran en evaluar el conocimiento y las habilidades adquiridas por el estudiante [22]. Como tal, el dominio de las habilidades de pensamiento computacional se basa en el análisis de componentes. Este enfoque es conveniente en la práctica, pero no necesariamente capta aspectos relacionados con el proceso de aprendizaje de los estudiantes. Estos enfoques generalmente se basan en preguntas de opción múltiple y de verdadero / falso, así como en preguntas abiertas.

Por ejemplo, Shell y Soh [20] utilizaron un conjunto de 26 preguntas conceptuales y de resolución de problemas para evaluar el desarrollo de las habilidades de pensamiento computacional. Las preguntas conceptuales se derivaron de un curso CS-1. Chen y col. [5] quería comprender el efecto que una actividad con un robot educativo tenía en las habilidades de pensamiento computacional de los estudiantes, por lo que desarrollaron 15 preguntas de opción múltiple y 8 preguntas abiertas, que se contextualizaron en situaciones cotidianas y en escenarios específicos de programación de robots. Basawapatna y col. [2] quería estudiar si los estudiantes detectaban ciertos patrones básicos asociados a las habilidades de pensamiento computacional al desarrollar videojuegos o simulaciones científicas utilizando una herramienta de creación de prototipos. Desarrollaron el cuestionario de patrones de pensamiento computacional para evaluar la capacidad de los participantes para reconocer, comprender y transferir patrones a diferentes contextos que no son de videojuegos. Weintrop y Wilensky [25] desarrollaron la Evaluación conmutativa, un conjunto de 28 preguntas de opción múltiple, cada una de las cuales muestra un programa corto, escrito de manera equivalente en lenguaje de bloques (Snap) o basado en texto (JavaScript). La idea era evaluar la transferencia de conceptos entre el texto y los lenguajes de programación basados en bloques. Basu y col. [3] desarrolló y validó una prueba para evaluar las habilidades de pensamiento computacional de los estudiantes de 4° a 6° grado. Las tareas de prueba no están acopladas a representaciones de programación específicas, lo que hace que el enfoque se centre en las prácticas de pensamiento computacional en lugar de las estructuras de programación, lo que permite que la prueba se utilice




en diferentes planes de estudio y diferentes lenguajes de programación. Varios autores han utilizado tareas de Bebras Challenge [6] para evaluar las habilidades de pensamiento computacional de los estudiantes, como [7], [14] y [15].

En el marco de un análisis exploratorio del impacto del programa TG se utilizó la prueba de pensamiento computacional desarrollada por Román-González et al. [18], que evalúa si los estudiantes pueden resolver problemas usando secuencias de instrucciones, iteraciones, condicionales y variables. La prueba tiene 28 preguntas de opción múltiple, donde los estudiantes deben seleccionar el conjunto de instrucciones que cumplen con la tarea requerida. Las instrucciones se presentan utilizando un lenguaje de bloques, similar a Scratch o Snap. Se eligió esta prueba porque está disponible públicamente, ha sido validada con estudiantes de los grados 5 al 10 y la prueba ya está en español, lo que significa que se puede aplicar directamente a los participantes de este estudio. Los autores de la prueba la validaron calculando los coeficientes de correlación entre las puntuaciones de la prueba pensamiento computacional y dos instrumentos más (Habilidades Mentales Primarias y la prueba RP30 de Resolución de Problemas), encontrando una alta correlación positiva entre las puntuaciones de la prueba pensamiento computacional y la capacidad de resolución de problemas de los estudiantes y correlaciones positivas pequeñas a moderadas para otras habilidades mentales.

TEST DE PENSAMIENTO COMPUTACIONAL
 Adaptado de Román-González, M., Pérez-Durozou, J. C., Jiménez-Fernández, C. (2014). 2016.

Nombre: _____ Edad: _____
 Email: _____
 Colegio: _____ Curso: _____

Instrucciones:
 El test está compuesto por 28 preguntas. Todas las preguntas tienen 4 opciones de respuesta, de las cuales solo una es correcta.
 A partir de que comienzas el test dispones de 45 minutos para hacerlo lo mejor que puedas. No es necesario que contestes a todas las preguntas.
 Antes de comenzar el test, verás a un 3 preguntas para que te familiarices con el tipo de preguntas que te irá encontrando, y en las que aparecerán los siguientes personajes:

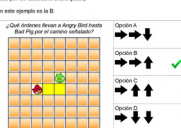
  

Ejemplo 1. En este primer ejemplo se te pregunta cuáles son los pasos que lleva a Angry Bird hasta Bad Pig por el camino señalado. En otros, llevar a Angry Bird EXACTAMENTE a la casilla en la que se encuentra Bad Pig sin pasarse ni quedarse corto, y siguiendo estrictamente el camino señalado en amarillo (con saltos y sin tocar los paredes, representadas por los cuadrados azules/grises).

La opción correcta en este ejemplo es la B.

¿Cuál camino lleva a Angry Bird hasta Bad Pig por el camino señalado?

Opción A	→ → → → →
Opción B	→ → → → →
Opción C	→ → → → →
Opción D	→ → → → →




Ejemplo 2. En este segundo ejemplo se te pregunta de nuevo cuál es la secuencia de pasos que lleva a Angry Bird hasta Bad Pig por el camino señalado. Pero en este caso las opciones de respuesta, en vez de ser flechas, son pasos que empiezan unos con otros.

Te recordamos que la pregunta te pide llevar a Angry Bird EXACTAMENTE a la casilla en la que se encuentra Bad Pig sin pasarse ni quedarse corto, y siguiendo estrictamente el camino señalado en amarillo (con saltos y sin tocar las paredes, representadas por los cuadrados azules/grises).

La opción correcta en este ejemplo es la C.

¿Cuál camino lleva a Angry Bird hasta Bad Pig por el camino señalado?

Opción A	→ → → → →	Opción B	→ → → → →
Opción C	→ → → → →	Opción D	→ → → → →




Ejemplo 3. En este tercer ejemplo se te pregunta que secuencia debe seguir el avión para dibujar la figura que aparece en la imagen. Es decir, cómo debe MOVER el lápiz para que se dibuje la figura.

El paso MOVER empuja el lápiz dibujando, mientras que el paso SALTAR hace jugar un salto al avión sin dibujar. La flecha gris indica la dirección del primer movimiento del lápiz.


La opción correcta en este ejemplo es la A.

¿Qué orden debe seguir el avión para dibujar la figura que aparece en la imagen?

Opción A	→ → → → →	Opción B	→ → → → →
Opción C	→ → → → →	Opción D	→ → → → →



1. ¿Qué secuencia lleva a Angry Bird hasta Bad Pig por el camino señalado?




A) → → → → →

B) → → → → →

C) → → → → →

D) → → → → →

2. ¿Qué pasos lleva en la secuencia para llevar a Angry Bird hasta Bad Pig por el camino señalado?



A) → → → → →

B) → → → → →

C) → → → → →

D) → → → → →

La Tabla 1 enumera las escuelas secundarias participantes, el número total de estudiantes por escuela, así como el tamaño promedio de la clase (“Clase promedio”). La columna “Maestro” indica cuántos maestros participaron en el programa TG en cada escuela, así como su género (M = Masculino, F = Femenino). Las escuelas más grandes tienen más de una clase por grado y la columna “Cohorte” indica el número estimado de estudiantes de 9º grado que participaron en el programa TG. Finalmente, las columnas “Entrada” y “Salida” indican cuántos estudiantes completaron la prueba pensamiento computacional, antes y después de completar el programa TG, respectivamente (la participación fue voluntaria). En el caso de la prueba de salida, también se indica el número de alumnas entre paréntesis, ya que esta información estaba disponible.

Escuelas	Estudiantes	Maestro	Cohorte	Entrada	Salida (mujeres)
INSUCO	850	1 F	200	14	41 (27)
LCD	230	1 F	80	45	63 (31)
LCVVH	340	1 F	80	30	22 (10)
				Total	126 (68)

Tabla 1: Escuelas participantes

Los tres establecimientos son escuelas técnicas, lo que significa que los estudiantes se gradúan con una carrera, como electricista, programador, asistente de contador, etc. Los maestros participantes completaron un curso de capacitación durante enero de 2020 y llevaron a cabo el programa TG durante el año escolar (marzo - diciembre de 2020). El programa TG se implementó durante la clase de Tecnología. Las escuelas LCD y LCVVH habían ofrecido el programa TG como una actividad extraescolar durante 2019, pero al igual que el resto de las escuelas, lo ofrecieron durante el horario escolar por primera vez en 2020.

Las escuelas se encuentran en ciudades que son capitales de provincia. A pesar de que sólo se trabajó con estudiantes de 9º grado durante 2020, las edades de los estudiantes oscilan entre los 13

y los 18 años, ya que estas escuelas profesionales también reciben a estudiantes que abandonaron las escuelas secundarias regulares. Para estas escuelas, este programa es probablemente su primer acercamiento a la informática.

La Figura 1 muestra las distribuciones de puntajes de las pruebas de entrada y salida por escuela, utilizando gráficas de densidad. Se ve que las escuelas LCD y LCVVH muestran alguna mejora en los puntajes de salida, en comparación con los puntajes de entrada. En la escuela INSUCO, se ve un desplazamiento hacia la izquierda en las puntuaciones de salida, lo que significa que los estudiantes en promedio obtuvieron puntuaciones más bajas en la prueba de salida. Por lo tanto, no está claro si estos puntajes representan una mejora en las habilidades de CT de los estudiantes.

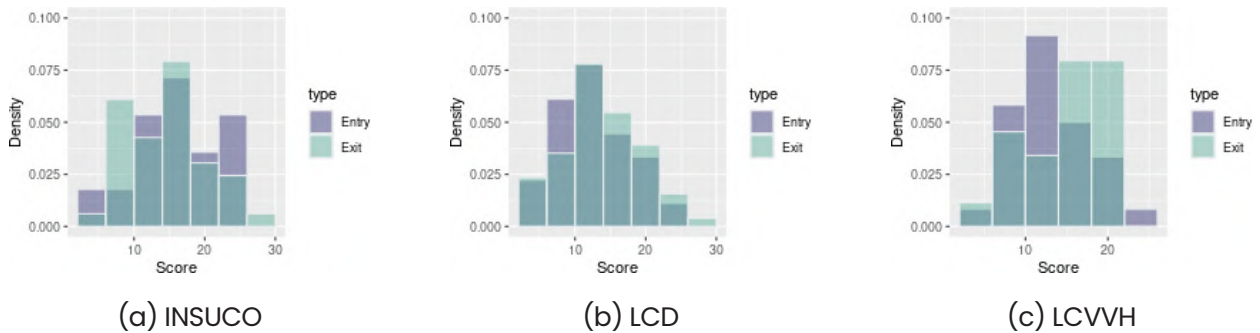


Figura 1: Gráficos de densidad de puntaje: pruebas de entrada y salida

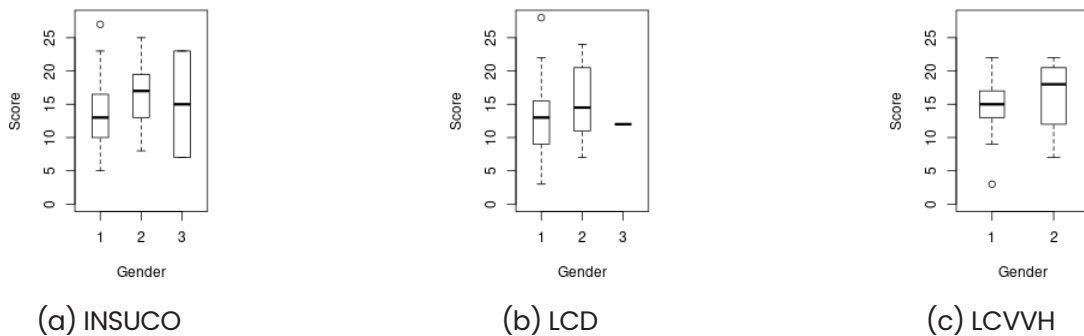


Figura 2: Puntajes de las pruebas de salida: distribución por género

Como los estudiantes indicaron su género durante la prueba de salida, en la Figura 2 se muestra la distribución de puntajes por género. Los estudiantes pueden ser mujeres (1), hombres (2) o no binarios / sin respuesta (3). Aquí se puede ver que el rango intercuartil para las mujeres en los establecimientos LCD y LCVVH es más menor que el de los hombres, lo que significa que las mujeres presentan menos variabilidad en la mitad media de los puntajes.

La pregunta es si los puntajes se distribuyen de manera similar al agrupar a los estudiantes por género. Los gráficos de la Figura 3 muestran la distribución de puntajes por escuela, tomando en cuenta el sexo y la edad.

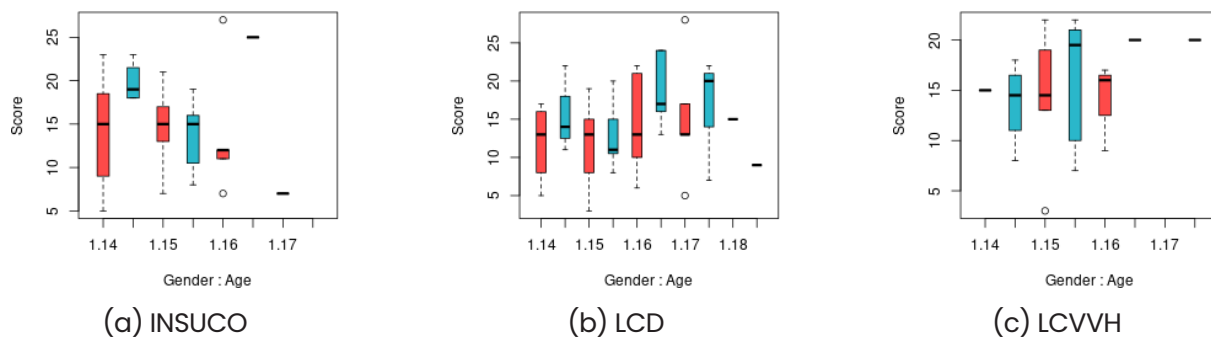


Figura 3: Puntajes de las pruebas de salida: distribución por género y edad

Dado que pocos participantes eligieron no binario / sin respuesta, se retiró a estos participantes de este análisis. Las etiquetas del eje X indican la combinación correspondiente (Sexo: Edad) para cada barra (por ejemplo, "1,14" significa participantes femeninas de 14 años). También se coloreó las barras por género para facilitar la lectura (rojo = mujeres, azul = hombres). En INSUCO, se ve que la mediana cambia por género para los participantes de 15 años, que también se observa en LCD. Finalmente, en LCVVH se ve que la mediana se mantiene estable o mejora, independientemente del sexo.

También se quiso contrastar los puntajes que obtuvieron los estudiantes con los puntajes que pensaban que obtendrían en la prueba de salida. Los gráficos de burbujas de la Figura 4 muestran

esta relación, donde la puntuación esperada está en el eje Y (1 = muy mala, 7 = muy buena) y la puntuación de la prueba está en el eje X (0 - 28 puntos). El tamaño de la burbuja indica cuántos estudiantes tienen la misma combinación (puntuación de la prueba, puntuación esperada) y el color de la burbuja indica el género. Las burbujas cerca de la diagonal principal significan que un participante obtuvo una puntuación similar a sus expectativas. Las burbujas sobre la diagonal son participantes que creían que les iría mejor en la prueba que cómo lo hicieron realmente, y debajo de la diagonal significa que lo hicieron mejor de lo esperado. Se ve que, en general, los estudiantes tenían confianza en sus respuestas, siendo los estudiantes de LCVVH los más pesimistas acerca de sus puntuaciones (independientemente del género). También se ve que en INSUCO los hombres eran más optimistas sobre su desempeño que las mujeres.

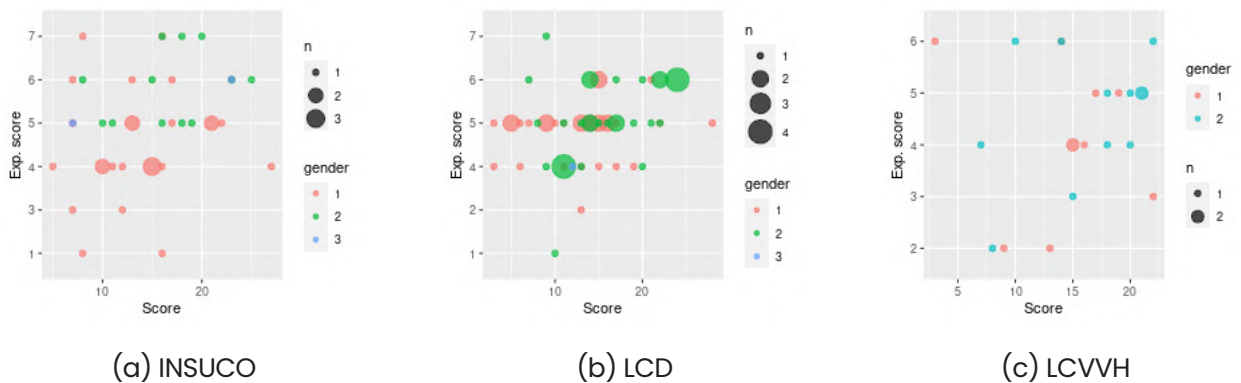


Figura 4: Puntaje esperado versus puntaje real, por género

Finalmente, se analizó el porcentaje de respuestas correctas por pregunta y escuela, antes y después de completar el programa TG. El gráfico de la izquierda en la Figura 5 muestra los porcentajes de la prueba de entrada, el gráfico de la derecha muestra los datos de la prueba de salida. Aquí se puede ver que ambas curvas son similares, aunque hay menos variabilidad en la prueba de salida, especialmente entre las preguntas 15 y 23. También se ve una ligera mejora en el porcentaje de respuestas correctas para las preguntas 12, 22 y 23. Sin embargo, también se ve que preguntas como la 8 y la 25 ahora tienen una tasa de respuesta incorrecta más alta, en todas las escuelas, por lo que una actividad en el programa TG puede estar causando cierta confusión entre los participantes, aunque no se puede descartar que puede haber problemas con la redacción de la pregunta que puede estar causando este efecto.

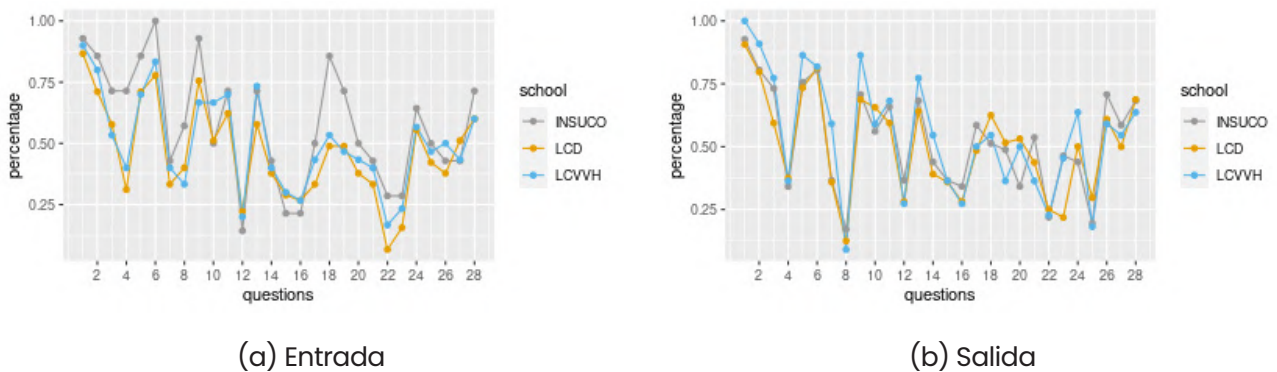


Figura 5: Porcentaje de respuestas correctas por pregunta, por escuela

DISCUSIÓN

Se vio una ligera mejora en los puntajes de las pruebas de salida, lo cual es prometedor dado que el enfoque principal de TG es la ideación del producto, el lanzamiento y la creación de prototipos, y no la programación. Los estudiantes utilizan una plataforma de código bajo (Thunkable) para desarrollar aplicaciones móviles, lo que aún requiere que piensen en los procesos lógicos y comerciales que necesitan implementar para materializar sus ideas. Los estudiantes también deben trabajar en la comunicación de aspectos técnicos de su trabajo, como decisiones de diseño e implementación.

Los equipos podrían ser de género mixto, pero se vio que los estudiantes varones generalmente se desempeñaron mejor que las estudiantes en la prueba de salida, por lo que las estudiantes pueden beneficiarse de actividades adicionales sólo para mujeres antes / durante la ejecución del programa TG en la escuela.

Adaptarse a la escuela en línea durante la pandemia de Covid-19 también fue un desafío. En América del Sur, los primeros casos de Covid-19 aparecieron en marzo de 2020, y las escuelas cerraron en abril debido a los temores de contagio. Las escuelas se apresuraron a hacer la transición a la enseñanza en línea, a medida que aumentaba el número de contagios y se extendía el bloqueo general. Los estudiantes tenían que adaptarse a los dispositivos que ya tenían sus familias, a veces compartiendo el mismo computador con varios hermanos para poder asistir a clase. Esto hizo que trabajar en línea fuera un desafío, ya que tanto los estudiantes como los profesores tuvieron que adaptarse a nuevas herramientas y dinámicas, sin mucho tiempo para prepararse. Algunas escuelas también tuvieron problemas con la asistencia, como en LCD y LCVVH, donde solo el 60% de los estudiantes se presentó a clases online, pero sólo el 20% era constante, el resto variaba de una semana a otra.

También se vio que el programa TG tuvo diferentes efectos en los estudiantes, dependiendo de las condiciones de la escuela, como la edad de los participantes y la distribución por género. Las escuelas analizadas son de diferentes tamaños: INSUCO es la escuela más grande en la cohorte de 2020, mientras que LCD y LCVVH son más pequeñas. Las tres escuelas tenían una maestra a cargo del programa TG, y hubo casi paridad entre los estudiantes masculinos y femeninos en la prueba de

salida. Sin embargo, se ve que las alumnas del primero medio en promedio obtuvieron puntuaciones más bajas que los alumnos varones, pero esto no es consistente por edad. Por ejemplo, tanto INSUCO como LCD tienen clases de tamaño similar (20-23 estudiantes), y aquí se ve que las estudiantes de 15 años tienen un desempeño similar o mejor que sus compañeros varones. Por otro lado, LCVVH tiene 28 estudiantes por aula en promedio, y se ve que los estudiantes varones superan a las estudiantes independientemente de la edad.

Aunque originalmente se había planeado realizar el programa TG en persona, se tuvo que improvisar debido a la pandemia de Covid-19. Se cree que estos resultados muestran que los cambios que se hicieron para adaptar el programa al contexto vivido en Chile fueron razonables y han tenido un efecto positivo en los estudiantes participantes. También se realizó un grupo focal al final del año escolar, donde la mayoría de los participantes indicaron que ahora entendían que crear una aplicación es mucho más que programar, es necesario identificar un problema real y luego entender cómo (y si) la tecnología se puede utilizar para resolver ciertos aspectos del problema.

Ninguno de los integrantes de la cohorte de 2020 logró el desafío internacional de TG, ya que las barreras de entrada para los estudiantes de las escuelas de oficios de un país que no habla inglés son simplemente demasiado altas. Sin embargo, tanto los profesores como los estudiantes quedaron satisfechos en general con la experiencia. De dos a cinco equipos por escuela completaron sus prototipos y los presentaron con éxito a un panel de jueces locales. En general, los participantes estaban seguros de lo que habían aprendido en el programa de TG, como lo demuestra el exceso de confianza que se vio en los gráficos de burbujas de la Figura 4, ya que no sabían que la prueba sólo evaluaba sus habilidades de pensamiento computacional, y no el conjunto de lo que aprendieron como parte del programa TG.

CONCLUSIONES

Se encontró que el programa TG se puede llevar a cabo durante el horario escolar en las escuelas secundarias mixtas que ofrecen formación profesional. Esto implicó hacer que el programa TG forme parte de una clase obligatoria, ampliar el plan de estudios TG existente, adaptar actividades y dinámicas a la enseñanza en línea y relajar los requisitos del idioma TG. También se encontró que el programa de TG tuvo un efecto leve y positivo en las habilidades de CT de la cohorte de 2020, aunque estos efectos dependen del género, la edad y las condiciones escolares. Se cree que las estudiantes pueden beneficiarse de las actividades exclusivas para mujeres antes y durante el programa TG, con el fin de aumentar su exposición a la tecnología, a fin de nivelar el campo de juego con respecto a sus compañeros masculinos.

Esta toma inicial y exploratoria que busca medir el impacto del programa TG en niñas y jóvenes nos muestra un camino para seguir explorando el impacto del programa, así como el determinar el impacto que el programa puede tener, en términos de adquisiciones de conocimientos en programación, se cree que hay que seguir estudiando a las alumnas, quizás con una herramienta que pueda ver en forma más integral la adquisición del conocimiento en torno a la programación.

REFERENCIAS

- [1] Fernando Alegre, John Underwood, Juana Moreno, and Mario Alegre. 2020. Introduction to Computational Thinking: A New High School Curriculum Using CodeWorld. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (Portland, OR, USA) (SIGCSE '20). Association for Computing Machinery, New York, NY, USA, 992–998. <https://doi.org/10.1145/3328778.3366960>
- [2] Ashok Basawapatna, Kyu Han Koh, Alexander Repenning, David C. Webb, and Krista Sekeres Marshall. 2011. Recognizing Computational Thinking Patterns. In Proceedings of the 42nd ACM Technical Symposium on Computer Science Education (Dallas, TX, USA) (SIGCSE '11). Association for Computing Machinery, New York, NY, USA, 245–250. <https://doi.org/10.1145/1953163.1953241>
- [3] Satabdi Basu, Daisy Rutstein, Yuning Xu, and Linda Shear. 2020. A Principled Approach to Designing a Computational Thinking Practices Assessment for Early Grades. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (Portland, OR, USA) (SIGCSE '20). Association for Computing Machinery, New York, NY, USA, 912–918. <https://doi.org/10.1145/3328778.3366849>
- [4] Heather Bort, Shion Guha, and Dennis Brylow. 2018. The Impact of Exploring Computer Science in Wisconsin: Where Disadvantage is an Advantage. In Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (Larnaca, Cyprus) (ITiCSE 2018). Association for Computing Machinery, New York, NY, USA, 57–62. <https://doi.org/10.1145/3197091.3197140>
- [5] Guanhua Chen, Ji Shen, Lauren Barth-Cohen, Shiyang Jiang, Xiaoting Huang, and Moataz Eltoukhy. 2017. Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & Education* 109 (2017), 162 – 175. <https://doi.org/10.1016/j.compedu.2017.03.001>

- [6] Valentina Dagiene and Gerald Futschek. 2008. Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks. In *Informatics Education - Supporting Computational Thinking*, Roland T. Mittermeir and Maciej M. Sysło (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 19–30.
- [7] Havva Delal and Diler Oner. 2020. Developing Middle School Students' Computational Thinking Skills Using Unplugged Computing Activities. *Informatics in Education* 19 (2020), 1–13.
- [8] Nora A. Escherle, Silvia I. Ramirez-Ramirez, Ashok R. Basawapatna, Dorit Assaf, Alexander Repenning, Carmine Maiello, Yasko Ch. Endo, and Juan A. Nolasco-Flores. 2016. Piloting Computer Science Education Week in Mexico. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (Memphis, Tennessee, USA) (SIGCSE '16)*. Association for Computing Machinery, New York, NY, USA, 431–436. <https://doi.org/10.1145/2839509.2844598>
- [9] Cheri Fancsali, Linda Tigani, Paulina Toro Isaza, and Rachel Cole. 2018. A Landscape Study of Computer Science Education in NYC: Early Findings and Implications for Policy and Practice. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE '18)*. Association for Computing Machinery, New York, NY, USA, 44–49. <https://doi.org/10.1145/3159450.3159467>
- [10] Brittany Terese Fasy, Stacey A. Hancock, Barbara Z. Komlos, Brendan Kristiansen, Samuel Micka, and Allison S. Theobald. 2020. Bring the Page to Life: Engaging Rural Students in Computer Science Using Alice. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (Trondheim, Norway) (ITICSE '20)*. Association for Computing Machinery, New York, NY, USA, 110–116. <https://doi.org/10.1145/3341525.3387367>
- [11] Fredrik Heintz and Linda Mannila. 2018. Computational Thinking for All: An Experience Report on Scaling up Teaching Computational Thinking to All Students in a Major City in Sweden.

In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE '18). Association for Computing Machinery, New York, NY, USA, 137–142. <https://doi.org/10.1145/3159450.3159586>

- [12] Helen H. Hu, Cecily Heiner, Thomas Gagne, and Carl Lyman. 2017. Building a Statewide Computer Science Teacher Pipeline. In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education (Seattle, Washington, USA) (SIGCSE '17). Association for Computing Machinery, New York, NY, USA, 291–296. <https://doi.org/10.1145/3017680.3017788>
- [13] Irene A. Lee, Maureen Psaila Dombrowski, and Ed Angel. 2017. Preparing STEM Teachers to Offer New Mexico Computer Science for All. In Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education (Seattle, Washington, USA) (SIGCSE '17). Association for Computing Machinery, New York, NY, USA, 363–368. <https://doi.org/10.1145/3017680.3017719>
- [14] Aidan Mooney and James Lockwood. 2020. The Analysis of a Novel Computational Thinking Test in a First Year Undergraduate Computer Science Course. *II Ireland Journal of Higher Education* 12 (02 2020).
- [15] Javier Olmo, Ramón Cózar, and José Antonio González-Calero. 2020. Computational thinking through unplugged activities in early years of Primary Education. *Computers & Education* 150 (02 2020), 103832. <https://doi.org/10.1016/j.compedu.2020.103832>
- [16] Patricia Ordóñez Franco, Joseph Carroll-Miranda, María López Delgado, Eliud Gerena López, and Grace Rodríguez Gómez. 2018. Incorporating Computational Thinking in the Classrooms of Puerto Rico: How a MOOC Served as an Outreach and Recruitment Tool for Computer Science Education. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE '18). Association for Computing Machinery, New York, NY, USA, 296–301. <https://doi.org/10.1145/3159450.3159544>

- [17] Chris Rhoton. 2018. Examining the State of CS Education in Virginia’s High Schools. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE ’18). Association for Computing Machinery, New York, NY, USA, 970–974. <https://doi.org/10.1145/3159450.3159492>
- [18] Marcos Román-González, Juan-Carlos Pérez-González, and Carmen Jiménez-Fernández. 2017. Which Cognitive Abilities Underlie Computational Thinking? Criterion Validity of the Computational Thinking Test. *Comput. Hum. Behav.* 72, C (July 2017), 678–691. <https://doi.org/10.1016/j.chb.2016.08.047>
- [19] Jean Salac, Max White, Ashley Wang, and Diana Franklin. 2019. An Analysis through an Equity Lens of the Implementation of Computer Science in K–8 Classrooms in a Large, Urban School District. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (Minneapolis, MN, USA) (SIGCSE ’19). Association for Computing Machinery, New York, NY, USA, 1150–1156. <https://doi.org/10.1145/3287324.3287353>
- [20] Duane F. Shell and Leen-Kiat Soh. 2013. Profiles of Motivated Self-Regulation in College Computer Science Courses: Differences in Major versus Required Non-Major Courses. *Journal of Science Education and Technology* 22 (2013), 899–913.
- [21] Jocelyn Simmonds, Francisco J. Gutierrez, Cecilia Casanova, Cecilia Sotomayor, and Nancy Hitschfeld. 2019. A Teacher Workshop for Introducing Computational Thinking in Rural and Vulnerable Environments. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (Minneapolis, MN, USA) (SIGCSE ’19). Association for Computing Machinery, New York, NY, USA, 1143–1149. <https://doi.org/10.1145/3287324.3287456>
- [22] Xiaodan Tang, Yue Yin, Qiao Lin, Roxana Hadad, and Xiaoming Zhai. 2020. Assessing computational thinking: A systematic review of empirical studies. *Computers & Education* 148 (2020), 103798.

<https://doi.org/10.1016/j.compedu.2019.103798>

- [23] Bishakha Upadhyaya, Monica M. McGill, and Adrienne Decker. 2020. A Longitudinal Analysis of K-12 Computing Education Research in the United States: Implications and Recommendations for Change. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (Portland, OR, USA) (SIGCSE '20). Association for Computing Machinery, New York, NY, USA, 605–611. <https://doi.org/10.1145/3328778.3366809>

- [24] Jayce R. Warner, Carol L. Fletcher, Ryan Torbey, and Lisa S. Garbrecht. 2019. Increasing Capacity for Computer Science Education in Rural Areas through a Large-Scale Collective Impact Model. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (Minneapolis, MN, USA) (SIGCSE '19). Association for Computing Machinery, New York, NY, USA, 1157–1163. <https://doi.org/10.1145/3287324.3287418>

- [25] David Weintrop and Uri Wilensky. 2015. Using Commutative Assessments to Compare Conceptual Understanding in Blocks-Based and Text-Based Programs. In Proceedings of the Eleventh Annual International Conference on International Computing Education Research (Omaha, Nebraska, USA) (ICER '15). Association for Computing Machinery, New York, NY, USA, 101–110. <https://doi.org/10.1145/2787622.2787721>

- [26] Hugh E. Williams, Selina Williams, and Kristy Kendall. 2020. CS in Schools: Developing a Sustainable Coding Programme in Australian Schools. In Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (Trondheim, Norway) (ITiCSE '20). Association for Computing Machinery, New York, NY, USA, 321–327. <https://doi.org/10.1145/3341525.3387422>

 **TECHNOVATION**
Girls Chile



MOTOROLA SOLUTIONS
FOUNDATION