

# **Videojuego educativo para enseñar algoritmos relacionados a grafos**

Alonso Utreras  
Profesor Guía: Iván Sipirán  
Departamento de Ciencias de la Computación  
Universidad de Chile  
Santiago, Chile  
`alonso.utreras@ug.uchile.cl`  
`autreras@dcc.uchile.cl`

Introducción

Hola holi holu

## **1 Trabajo relacionado**

### **1.1 Propiedades que debería tener el juego**

## **2 Problema**

## **3 Preguntas de Investigación**

## **4 Hipótesis**

## **5 Objetivo principal**

Diseñar y desarrollar un videojuego que muestre grafos y algoritmos relacionados a estos, que permita al usuario ir ejecutando acciones en la aplicación según lo pide un algoritmo desplegado en la misma pantalla. La aplicación debe indicar al usuario cuando está en lo correcto y cuándo se equivoca.

## **6 Objetivos específicos**

- Diseñar una aplicación interactiva que muestre grafos y permita seguir los pasos relacionados a algoritmos que trabajen con grafos.
- Medir el interés de estudiantes de computación relacionado a grafos antes y después del uso de esta herramienta.
- Medir la comprensión y capacidad de replicación de un algoritmo relacionado a grafos de un grupo de estudiantes de computación antes y después de probar el videojuego presentado.

## 7 Metodología

Se desarrolla un videojuego considerando el feedback de un profesor guía y otros estudiantes que no sean parte de la población objetivo del estudio, ajustándolo según sea necesario.

Asegurarse que el videojuego cumple con las características seguras por autores que proponen metodologías para la evaluación de videojuegos educativos: Gibson et al. [?], Petri, Giano y otros, [?] y Kiili et al. [?]. Si no, seguir iterando.

Cuando se haya impartido la materia relacionada a grafos y se hayan visto los algoritmos BFS y DFS, preguntarle a los alumnos si se creen capaces de programar tales algoritmos desde 0 y guardar estos resultados. Presentar el juego en los cursos cuando se esté impartiendo la materia relacionada a grafos. El uso del videojuego será voluntario y sugerido, con el requisito de llenar un formulario posterior al uso del juego. El formulario poseerá un cuestionario siguiendo los lineamientos de MEEGA+ planteados [?] Se revisarán los datos llenados por los estudiantes que accedieron al formulario, en conjunto con los datos recolectados durante el juego mismo, para ver si estos se condicen. Por ejemplo, el juego medirá el tiempo entre clicks. Se espera que un jugador que esté motivado posea un alto nivel de actividad, y una alta cadencia de clicks.

Finalmente, se realizará una prueba de programación en los cursos que vean la materia de grafos, donde los estudiantes tendrán que responder preguntas relacionadas a los algoritmos vistos en el juego. Se separarán dos grupos, uno de control con la gente que decidió no jugar, y otro experimental que sí probó el videojuego.

El programa ofrecerá a los jugadores ejecutar tareas relacionadas con los algoritmos BFS, DFS, Kruskal y Prim relacionados a grafos. Para completar los desafíos presentados, el usuario deberá realizar las instrucciones dictadas por un computador, invirtiendo los roles de un programador y un computador. En este caso, el juego limita al jugador por medio de sus reglas, a seguir los pasos del algoritmo presentado. Si el jugador ejecuta correctamente las instrucciones, ganará, pudiendo pasar a otro nivel.

El videojuego guardará datos durante su ejecución y los enviará a alguna base de datos donde se almacenarán para su posterior análisis. Los datos que se guardarán serán: clicks, movimientos del mouse, acciones del teclado a lo largo del tiempo. Estos datos se usarán para contrastar con la información final y para determinar el grado de veracidad de las respuestas.

### 7.1 Desventajas y debilidades del proceso

El tamaño muestral puede no ser suficiente. No se sabe cuánto será el tamaño de los grupos de control y experimental, pero se espera que el total de participantes sea superior a 50.

Existe un sesgo al presentar el uso del videojuego como una opción voluntaria, por lo que los estudiantes con más tiempo e interés tenderán a probar el videojuego, pero son quienes a su vez tienen mejores notas.

## 8 Resultados esperados

## 9 Conclusiones

### 9.1 Trabajo realizado

### 9.2 Discusión

### 9.3 Trabajo futuro

\*

## References

- [1] *CodeCombat: Coding games to learn python and JavaScript*. <https://www.codecombat.com/>. Accessed: 2022-May-20.
- [2] *Scratch: Learn Programming with blocks*. <https://scratch.mit.edu/>. Accessed: 2022-May-20.
- [3] Reham Ayman, Nada Sharaf, Ghada Ahmed & Slim Abdennadher (2018): *MiniColon; teaching kids computational thinking using an interactive serious game*. In: *Joint International Conference on Serious Games*, Springer, pp. 79–90.
- [4] Sasha A Barab, Brianna Scott, Sinem Siyahhan, Robert Goldstone, Adam Ingram-Goble, Steven J Zuiker & Scott Warren (2009): *Transformational play as a curricular scaffold: Using videogames to support science education*. *Journal of Science Education and Technology* 18(4), pp. 305–320.
- [5] Louis Deslauriers, Logan S McCarty, Kelly Miller, Kristina Callaghan & Greg Kestin (2019): *Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom*. *Proceedings of the National Academy of Sciences* 116(39), pp. 19251–19257.
- [6] Michael Eagle & Tiffany Barnes (2009): *Experimental evaluation of an educational game for improved learning in introductory computing*. *ACM SIGCSE Bulletin* 41(1), pp. 321–325.
- [7] Sarah Esper, Stephen R Foster, William G Griswold, Carlos Herrera & Wyatt Snyder (2014): *CodeSpells: bridging educational language features with industry-standard languages*. In: *Proceedings of the 14th Koli calling international conference on computing education research*, pp. 05–14.
- [8] N Fraser (2013): *Blockly*. Available at <https://developers.google.com/blockly/>:Google.
- [9] Carlo Ghezzi, Mehdi Jazayeri & Dino Mandrioli (1991): *Fundamentals of software engineering*. Prentice-Hall, Inc.
- [10] Andreas Giannakoulas & Stelios Xinogalos (2018): *A pilot study on the effectiveness and acceptance of an educational game for teaching programming concepts to primary school students*. *Education and Information Technologies* 23(5), pp. 2029–2052.
- [11] Ben Gibson & Tim Bell (2013): *Evaluation of games for teaching computer science*. In: *Proceedings of the 8th Workshop in Primary and Secondary Computing Education*, pp. 51–60.
- [12] Lindsey Ann Gouws, Karen Bradshaw & Peter Wentworth (2013): *Computational thinking in educational activities: an evaluation of the educational game light-bot*. In: *Proceedings of the 18th ACM conference on Innovation and technology in computer science education*, pp. 10–15.
- [13] Susanna Hartikainen, Heta Rintala, Laura Pylväs & Petri Nokelainen (2019): *The Concept of Active Learning and the Measurement of Learning Outcomes: A Review of Research in Engineering Higher Education*. *Education Sciences* 9(4), doi:10.3390/educsci9040276. Available at <https://www.mdpi.com/2227-7102/9/4/276>.

- [14] Kristian Kiili, Keith Devlin, Arttu Perttula, Pauliina Tuomi & Antero Lindstedt (2015): *Using video games to combine learning and assessment in mathematics education*. *International Journal of Serious Games* 2(4), pp. 37–55.
- [15] John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman & Evelyn Eastmond (2010): *The scratch programming language and environment*. *ACM Transactions on Computing Education (TOCE)* 10(4), pp. 1–15.
- [16] Jane McGonigal (2011): *Reality is broken: Why games make us better and how they can change the world*. Penguin.
- [17] Giani Petri, Christiane Gresse von Wangenheim & Adriano Ferreti Borgatto (2018): *MEEGA+: A method for the evaluation of educational games for computing education*. *INCoD–Brazilian Institute for Digital Convergence*, pp. 1–47.
- [18] Giani Petri et al. (2018): *A method for the evaluation of the quality of games for computing education*.
- [19] Christina Tikva & Efthimios Tambouris (2021): *Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature Review*. *Computers & Education* 162, p. 104083.
- [20] David Weintrop & Uri Wilensky (2012): *RoboBuilder: A program-to-play constructionist video game*. In: *Proceedings of the constructionism 2012 conference*. Athens, Greece.
- [21] Jeanette Wing (2011): *Research notebook: Computational thinking—What and why*. *The link magazine* 6, pp. 20–23.
- [22] G. Wurman Ventura (2013): *Desarrollo de videojuego de ejercitación aritmética para un computador interpersonal usando un mouse por niño en la sala de clases*. Master’s thesis, Pontificia Universidad Católica de Chile, doi:<https://doi.org/10.7764/tesisUC/ING/2876>. Available at <https://repositorio.uc.cl/handle/11534/2876>.
- [23] Daniela Zehetmeier, Axel Böttcher, Anne Brüggemann-Klein & Veronika Thurner (2019): *Defining the competence of abstract thinking and evaluating CS-students’ level of abstraction*. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences*.
- [24] Weinan Zhao & Valerie J Shute (2019): *Can playing a video game foster computational thinking skills?* *Computers & Education* 141, p. 103633.