A Game-Changing Instructor Tool to Reinforce Coding Concepts

Devorah Kletenik
kletenik@sci.brooklyn.cuny.edu
Brooklyn College, City University of New York

Deborah Sturm
Deborah.Sturm@csi.cuny.edu
College of Staten Island, City University of New York

ABSTRACT

We developed a 3D serious game that teaches and reinforces programming concepts. An innovative feature allows instructors to create customized challenges that students solve in the context of the game, allowing instructors to use the game to effectively target course topics or skills. A web-based portal gives instructors access to data about student performance in the game.

CCS CONCEPTS

• **Social and professional topics** → **CS1**; • **Applied computing** → **Computer games**.

KEYWORDS

introductory coding, CS1, serious games, game-based learning

1 INTRODUCTION

A number of *serious games* have been created to help teach introductory programming (e.g. [1, 3]). Based on research that suggests that serious games excel at both motivating students to learn in addition to effectively teaching subject matter, these games offer a new way to encourage students to practice coding. These initiatives are particularly needed for the introductory programming course, which suffers from high attrition rates of 30% – 50% (e.g. [2]), a rate which is often significantly higher for females/racial minorities.

A specific drawback to existing serious games for programming is that they offer fixed programming challenges. Instructors who teach introductory programming may prefer to hone in on specific topics, or to challenge students with particular types of programming problem. These games do not afford this flexibility. Our innovative serious game addresses this problem by allowing instructors to customize the coding challenges. A complementary web-based portal, under development, will give instructors access to detailed data about their students' gameplay.

2 OVERVIEW

Code Control is a 3D computer game that was developed using the Unity platform to engage, motivate and improve the learning experience for students taking introductory programming courses. The storyline centers around endangered animals who are missing from an animal rescue. The animals' digital name tags contain corrupted code that players must solve in order to save the animals.

For each programming problem, instructors submit directions, template code with areas for students to fill in, and a correct solution. When players find a lost animal in the game, a programming challenge pops up. Students are prompted with the directions and given a code window to fill in the missing code. Student's solutions are compiled and checked for correctness by comparing them to the instructor's solution. Compiler errors, execution errors, and/or positive feedback are provided along the way. If the student's solution is correct, the animal is saved.

We use the Judge0 API[1] to provide the automated code compilation and execution environment, and customized scripts to provide syntax highlighting. Judge0 supports 42 programming language, as well as multi-file programs, so *Code Control* can be used in a variety of courses at all levels.

A database connection allows us to collect detailed analytics of all users' gameplay (e.g. time spent per challenge, successes and mistakes made, levels completed, score, etc.). These data are displayed in a web portal for instructors so that they can use the game as an informal evaluation mechanism. *Code Control* can be played in a browser without requiring installation.

3 CONTRIBUTIONS AND FUTURE WORK

Our game is a useful and innovative way to encourage students to practice programming by solving customized challenges that accurately target their course goals. Our planned future work includes future development of the instructor portal, making the game more flexible to allow instructors to pose non-programming challenges (e.g. tracing, Parson's puzzles), and improving the game UI and gameplay. We also plan to conduct a comprehensive evaluation of its effectiveness. We are making *Code Control* freely available to instructors and welcome feedback about its use.

REFERENCES

[1] Tiffany Barnes, Heather Richter, Amanda Chaffin, Alex Godwin, Eve Powell, Tiffany Ralph, Paige Matthews, and Hyun Jordan. 2007. Game2Learn: A study of games as tools for learning introductory programming concepts. *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education* 7 (2007).
[2] Päivi Kinnunen and Lauri Malmi. 2006. Why Students Drop out CS1 Course?. In *Proceedings of the Second International Workshop on Computing Education Research (ICER '06).*
[3] Michael J Lee and Amy J Ko. 2011. Personifying programming tool feedback improves novice programmers' learning. In *Proceedings of the seventh international workshop on Computing education research.* ACM, 109–116.

[1] https://api.judge0.com/