Game Design as an intro to Computer Science

Mark Suter
HS Computer Tech teacher
Pandora, Ohio
Outline:

WHAT: Leveraging current student interests to expose more students to the field of computer science.

Mark Overmars article: referencing inheritance (parenting), object oriented thinking

HOW: Player and programmer agency, immediate, visual

10 min Lesson in GM:S in sir clicks-a-alot lvl3_moving
   - declare a variable: “can_shoot” (var types: boolean, int, string)
   - debug “obj_alarm3drops_done” only dropping 1 apple (loops, for statement)

Transferring DnD to GML, JS, C
   emphasize concepts over syntax
Teaching Computer Science through Game Design

Mark Overmars, Utrecht University

LOGO

LOGO is a programming language that is easy to learn and provides a fun way to teach computer science. It is based on a simple language called LOGO, which was created by the computer scientist, Seymour Papert. LOGO is a great tool for teaching programming concepts to students at all levels. It is easy to use and allows students to experiment with programming concepts in a fun and interactive way.

GAME DESIGN

Creating computer games is a popular activity for young people. Many students are interested in game design as a career. However, many students are not sure how to get started. Game design can be a challenging field, but it is also very rewarding. If you are interested in game design, there are many resources available to help you get started.

Users who can't code can use GameMaker to build games and learn object-oriented design.

Why not try programming as a vehicle to teach your students computer science?

Developing computer games involves many aspects of computing, including computer graphics, artificial intelligence, human-computer interaction, security, distributed programming, simulations, and software engineering. Game development also brings into play aspects of the liberal arts, social sciences, and psychology.

Creating a state-of-the-art commercial computer game is an extremely difficult task that typically requires a multidisciplinary budget and a development team that includes all of the above talents.
Teaching Computer Science through Game Design

Mark Overmars, Utrecht University

As a youth taking my first steps toward learning how to program, I was fascinated by games. The quest to learn more about them and how to build them became a central focus of my teenage years. Admittedly, in the process, I also learned a lot about computer science, which as a young computer scientist, I was not so keen on. I thought it was mostly about fixing bugs and writing boring code. 

I've since learned that computer science encompasses much more than that. It's not just about programming and debugging; it's about understanding the logic of the world, how to design systems, and how to solve complex problems. It's about making things work and making them work better. And, as I've grown older, I've come to realize that the excitement of creating a game is only matched by the thrill of making something that works and that is enjoyed by others.

In light of this, why not use game design as a vehicle to teach youngsters computer science? Developing computer games involves many aspects of computing, including computer graphics, artificial intelligence, human-computer interaction, security, distributed programming, simulation, and software engineering. Game development also brings into play aspects of the liberal arts, the social sciences, and psychology.

Creating a state-of-the-art commercial computer game is an incredibly difficult task that typically requires a multimillion-dollar budget and a development team that includes 40 or
Resources

“Teaching Computer Science through Game Design” by Mark Overmars, 2004

Gamemaker DnD to GML reference

GameMaker Tutorials (by Mr. Suter): https://sites.google.com/site/mrsuterstechlab/videos/tutorials

GameMaker Sir-Clicks-A-Lot Starter file. (You must IMPORT this, not OPEN it) http://1drv.ms/1jow9Ov