



#coding7 #scratch #computerscience #arcadegames

QUESTION

Depth of Knowledge Chart

How can I...

...UTILISE the provided sprites and code (if needed), to CODE my own recreation of this earliest arcade game, or use it's gameplay elements to create a "franken-game" creatively skinned into a different genre?

*I don't need to "Get a Life."
I'm a gamer. I have LOTS of lives.*

TASK

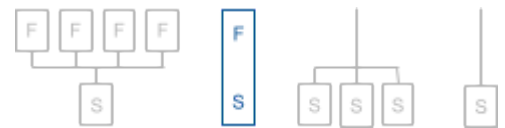
[F] Start here:

- ❑ Watch the video Computer Space Gameplay and read the gameplay section on the Wikipedia entry. (both at right)
- ❑ Make sure you understand the game before you begin to code.

[S] Code this game, or your "franken-game" (see Question above).

- ❑ Start by "remixing" the Velocity Code (at right) that you are using.
- ❑ Create ALL Sprites, and give them meaningful names. (you may need 6-7 sprites including a Control Sprite)
- ❑ Create ALL the variables you need. (HINT: for ALL sprites)
- ❑ The "Follow the Dot" code in the "I'm stuck" (at right) may help you in coding the movement for enemy1. (see below for enemy 2)
- ❑ Code the shooting code for the player, then enemy shooting.
- ❑ Code the *playerScore* and *enemyScore*, then the *timer*. (last) (it might be easier to code the scores while doing the shooting)
- ❑ You do not need to include the horrible seventies sounds.

(the above list is a guide only, it is not meant to be a complete step-by-step)



ANALYSIS

[read] [Computer Space @ Wikipedia](#)[watch] [Computer Space Gameplay](#)

[theory] Naming/Identifier conventions
[theory] Internal documentation
[theory] Software development lifecycle
[theory] Machine instruction cycle

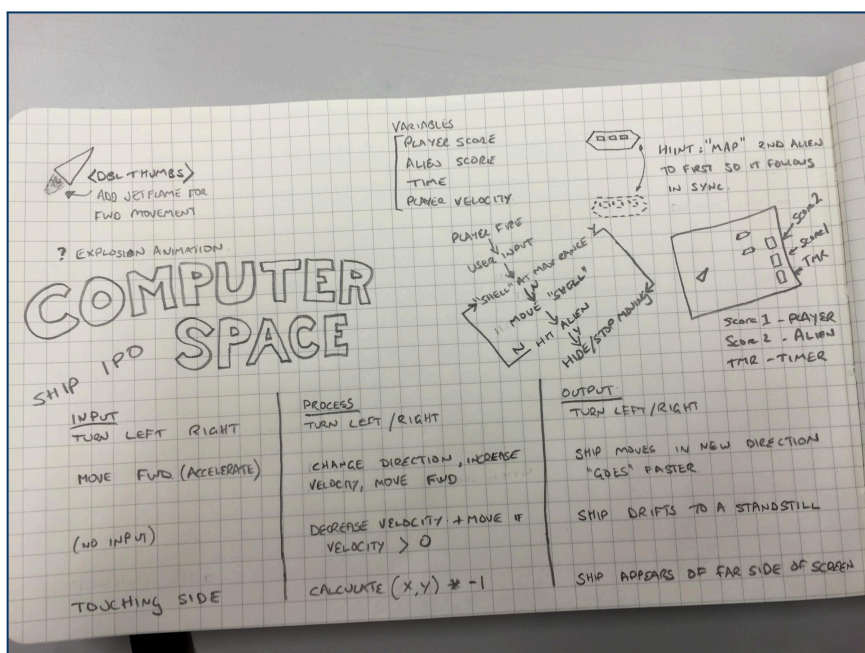
RESOURCES

[sprites]

[template]

[code] [Simple Spaceship Velocity](#)[code] [Realistic Spaceship Velocity](#)

DESIGN



(click image to enlarge)

I'M STUCK

[hint] [Follow the Dot](#)[check] [Assessment Notes](#)

VERSION 2.0

[think]

[modify]

[extend]

HINT: For the two enemies, have one simply follow the other one around. This may provide a hint for the code for *sprEnemy2*. It works best if the second enemy is its own Sprite (object) and not a Clone (instance).



ASSESSMENT NOTES

- ☐ The focus on this task is good programming style.
- ☐ .

FREQUENTLY ASKED QUESTIONS

Q. Which Velocity Should I use? (Simple or Realistic)

A. Either. It really is up to you, remembering that Simple Velocity has two Sprites for the Player, while Realistic Velocity has only one. It is a good idea to REMIX the one you choose rather than using your backpack and previous utilised code.

Q.

ACADEMIC STANDARDS & BENCHMARKS

[assessment notes](#) | [learning standards](#)

- ☐ 1.1 Students create original works or responsibly repurpose or remix digital resources in new creations.
- ☐ 2.1 Students communicate information and maintain a digital portfolio of their work.
- ☐ 2.2 Students will provide adequate instructions within the scope of the user experience.
In addition Coding 7 & Coding 8 students will be expected to include internal code documentation/comments to explain how the components of the algorithm function.
- ☐ 3.1 Students know and use a deliberate design process for generating ideas, testing concepts and creating programming solutions.
- ☐ 3.2 Students develop, test and refine code as part of a cyclical design process (software development lifecycle)
- ☐ 4.1 Students will use a development environment to select and combine control structures while writing modular code, using conventional casing and naming practices, to ensure code is consistent and readable.
- ☐ 4.2 Students will select and apply boolean expressions with the appropriate control structures to write logically correct programs.
- ☐ 4.3 Students can effectively declare and initialise the appropriate primitive data types and constants.
- ☐ 4.4 Students understand and use technology systems.



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License](#).
Based on a work at [aselley.com](#)
One Page Learning [[template here](#)]