

## **ICS3U1 FINAL PROJECT PLANNING**

For your final project in this course, you will be writing a large program in Java. You need to show that you've learned something more than the straight forward things that we did in class (e.g. a ball bouncing off of a paddle, a bigger tictactoe). You should be able to take a complicated idea/game and break it down into parts so that you can program it. You can use imagination and creativity to make your own version of something, or you can program an existing game.

You could

- ~~make a board game~~ (see <http://quarkphysics.ca/ICS3U1/unit4/BoardGame.html> for ideas) **\*\* not for June 2022 \*\***
- make an animation game (e.g. breakout, space invaders, asteroids, etc).
- something else (e.g. a paint program, a card game like Blackjack or Korean Go-Stop Game)

Please do not take something that you've already done and make it more complicated.

### ***Preliminaries***

- Name of game: \_\_\_\_\_
- Rules / How to play (write them out)
- Are you working alone or with a partner? \_\_\_\_\_

You must get the teacher to approve your game/program choice before you can work on it.

Some games are too hard: e.g. Pacman and possibly Battleships.

However, you may be able to work on a hard game and just get part of it done – eg. Chess or checkers.

However, Go and Othello are easy enough that you should be able to get the whole game working just fine.

### ***Planning***

Write out a detailed description of your game. If needed, you can add more things to this as you go along. Write out a plan for how your game will work and how you will program it. See the two attached examples. Show the teacher your detailed plan before you spend a lot of time programming it.

***Here's an example of part of the planning:***

What does your program have to do to win? How do you check for this based on how your data is stored?

*Example: Space Invaders has enemies and to win at Space Invaders you have to kill all of the enemies. I then need to explain how enemies are represented/stored in my program. Next I need to explain what happens to the data when an enemy is killed (what variable changes and how). Finally I need to explain how I would check if all enemies are killed.*

***Program***

Your program must meet all of the formatting and style requirements of the document  
<http://quarkphysics.ca/ICS3U1/unit2/require.htm>

## General

- Your game should have more than one screen (it could have a start screen and then the game, or multiple levels)
- Images are nice, but not mandatory. Sound effects are not hard to add.
- The user should be able to play again (ie. you don't have to exit the program and restart it in order to play again).
- The game should be playable:
  - o the UI should make sense
  - o the game is not so hard that it's impossible and not so easy that it's boring.
- The graphics should be attractive
- In many games you can add some more complexity. For example, you can't really make Connect 4 more complicated, but you can make "Breakout" more complicated by having some bricks dropping powerups (eg. slow ball) or "powerdowns" (eg. shrink paddle).
- If your game can have more than one level (e.g. as in most animation games) then you should try to implement this.
- Your game should be broken up into methods – each of which does one particular task, and probably also into different classes.

## Animation Game

- use an array or an array list
- 

## Board Game

- ?

## Other

- if you work with a partner, record what each person does as you go along
- Don't just copy something from online. It's normally quite easy to tell if someone cheats. (It's more obvious the more complex a program gets). If you can't figure out how to do some part of your game, check with the teacher. If you then get permission to copy some algorithm from online, make sure to document it in your notes & program.
- ~~Because often a lot of time is wasted when students do their final project, you will need to complete a daily tracking sheet. You need to write down what you are working on each day, what you have accomplished or what you need to figure out. [\*\*\* No, at Central SS there is a better work ethic]~~

## Marking

Your mark is based on

- how well the program is written
- how well it works
- how complex it is (a ball bouncing off of a paddle is not going to get a good mark).