ICS3UI Final project

For your final project, you will design a computer game and program it in Python.

You will complete your project in stages over the next two weeks and turn in each piece as you finish it, but the final turn-in date is Friday, June 13.

# Learning goals

Doing this project will help you to:

* Tie together everything you’ve learned this semester about algorithms, software design, if-statements, loops, arrays, functions, graphics and animations.
* Practice the art of stepwise refinement
* Learn the basics of GUI design: making buttons, and acting on the user’s mouse movements and key strokes
* Have fun while learning some cool new software design techniques!

# Marking

The final project is worth 15% of your report card mark. It is one half of your summative. (The other half is your Final Test.)   
  
See the Final Project Marking Form.xls for how marks are allocated.

# Turn-in schedule

|  |  |  |
| --- | --- | --- |
| **Due date** | **What you will turn in** | **Description** |
| **Fri, May 30** | Design Document 1st draft | Must be turned in through Edmodo before you begin coding in class |
| **Fri, June 6** | Version 1 of your game | Should have the complete runGame() procedure, with shell definitions of all the procedures that it calls. |
| **Fri, June 13** | Final version of your game + final Design Document | Should be a fully functioning game that meets the description of your design document. |

# Requirements for the game

You may implement a known game (such as Battleship) or make up your own game. Choose a game that you could comfortably finish in 10 class periods plus 8-10 hours of coding at home.

### Your game must have:

* An objective or challenge for the user (e.g. dodge things, collect things, kill things, solve a puzzle, race against the clock, etc.)
* A graphical screen that gets updated as the game progresses
* Mouse interaction of some kind
* Key-stroke interaction of some kind (e.g. ‘P’ for pause, arrow keys, etc.)
* Game statistics that get displayed on the game screen using create\_text (e.g. score, timer, lives left, etc.)
* Animation of some kind

### Your Python code must have:

* A main procedure at the bottom of the file named **runGame()**, which gets called once at the start of the program and uses a while-loop to run the game.
* At least four other functions or procedures besides runGame. For example, **drawCurrentBoard()**, **updatePlayerPosition(), updateScore(),** etc.   
  The procedure **runGame()** would make use of these other procedures.
* All of its game code contained in functions or procedures. There should be no bare code between functions or at the end, except for the code at the top that sets up the screen and the code at the bottom that binds user-actions to the procedures.
* Ample documentation, including a heading. There should be enough red comments so that someone not familiar with your project could easily follow your code. The sample game **Hiker And The Bees Game.py** is a good example to follow, mostly because I wrote it.

# Requirements for the design document (see sample)

The design document must contain the following:

1. A 100-150-word description of the game. Describe the initial set-up, the object of the game, the rules, how score is kept, etc. Assume your reader has never heard of the game, even if it’s a well-known game such as Hangman.
2. A snap-shot drawing of what the screen would look like in a typical game. You may create your drawing using Word, Paint or Photoshop, OR you may simply draw it on paper, take a photo of it and paste the photo into the design document.
3. Instructions for playing the game (how to start, which button does what, etc.)
4. A pseudocode algorithm for:
   1. your **runGame** procedure
   2. your mouseClick or mouseMotion procedure

# One final remark

As you code, you’ll probably get ideas for new rules and new features to add, ones you didn’t think of when you were writing the first draft of your design. That’s okay! Go ahead and add them! Just make sure that the final design document you turn in matches the game that you turn in.