CS 175 | Scripting Languages
Programming Assignment 4

This assignment is about focusing on writing your own object manager and simple collision handling. The estimated time for completion is about six hours.

The program given to you is a continuation of assignment 3 (Game State Manager), and you'll have to implement the object manager. You will be given all the needed assets and classes (as files) except for the following, which you will have to implement or edit:

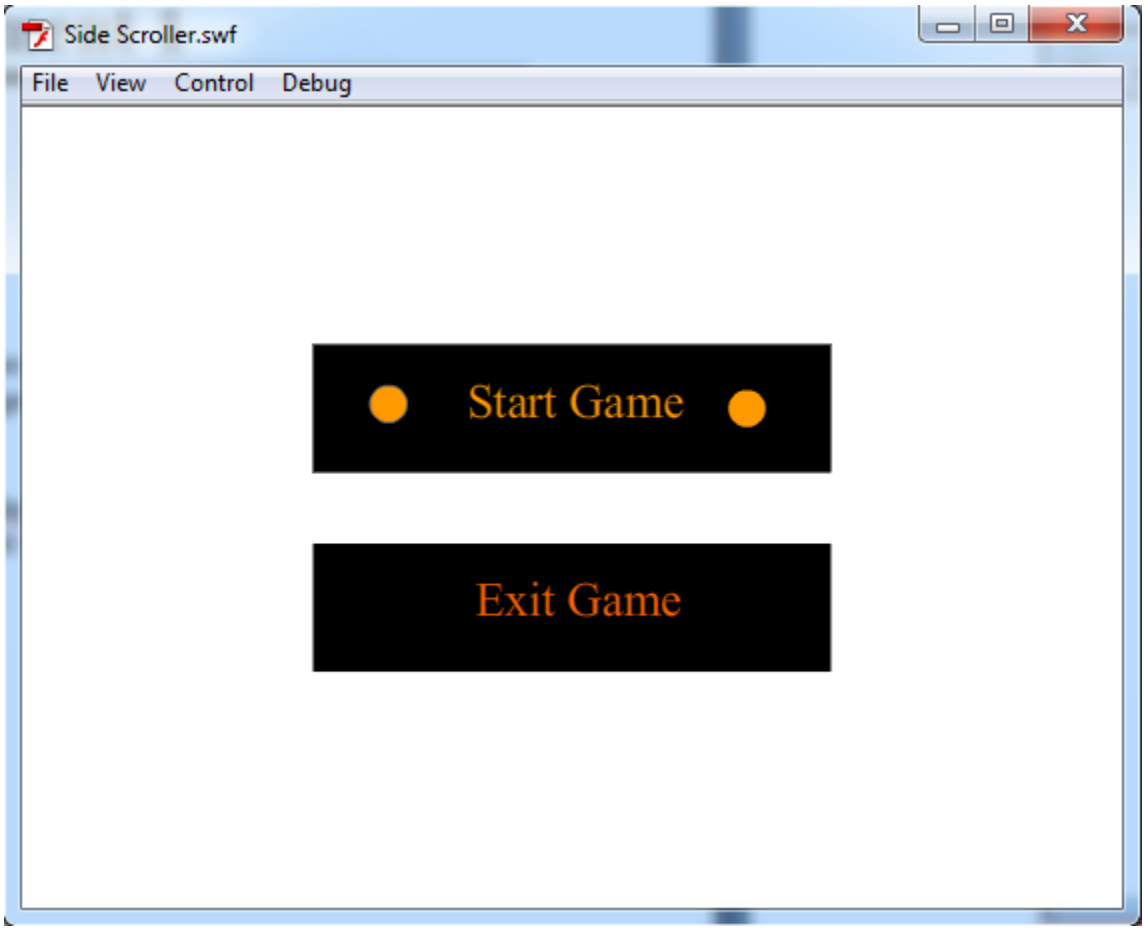
- **GameStateManager.as**
- **ObjectManager.as**

Details

Side Scroller Details:

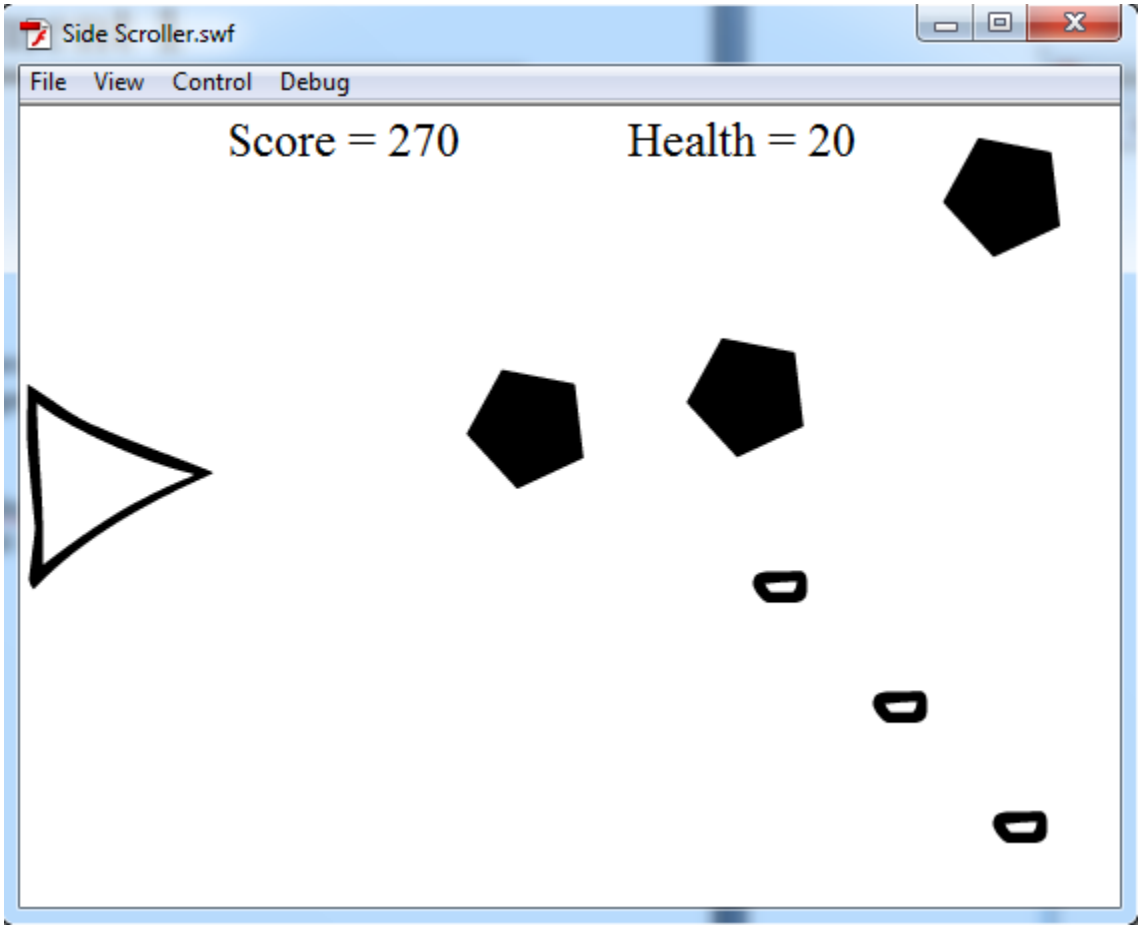
The game has two levels already implemented for you (MainMenu and Level1). Below is a description of their behavior.

Main Menu



- ***In the MainMenu level, two MovieClips have to be shown (StartGame and ExitGame)***
- ***The player uses the "UP" and "DOWN" keys in order to change the selection between the two.***
- ***Pressing the "SPACE" key will switch the game to a different level, depending on the user's selection.***
- ***Pressing the "R" key anytime during the game will restart the MainMenu***

Level 1



- ***Level 1 is a simple shooter***
  - ***The level starts with a simple countdown (no enemies are generated until the countdown is over).***
  - ***The ship goes up and down when user presses the "UP" or "DOWN" keys.***
  - ***The ship shoots bullets when the user presses the "SPACE" key.***
  - ***Enemies (black stars) get generated on the right and move towards the ship***
  - ***Enemies and Bullets are destroyed if***
    - ***they collide, which increases the score***
    - ***they are outside the screen (since we don't have use for them anymore)***
  - ***The ship's health is decreased if hit by an enemy.***
  - ***The game ends if the ship's health reaches 0 by going back to the main menu***
- ***Pressing the "M" key anytime during the game will take us back to the main menu***
- ***Pressing the "R" key anytime during the game will restart the level***

PS: Check the given swf file for more details or to play the game

Code Details:

In this assignment we will be adding a new component (Object Manager) to the game engine. The current "Engine" folder contains the following classes **(Found in the Engine Folder)**:

- Game.as
- GameObject.as
- GameStateManager.as (for you to edit)
- InputManager.as
- ObjectManager.as (for you to implement)
- State.as

Using the above classes, the following classes are created **(Found in the GamePlay Folder):**

- Inherited from the State Class
  - MainMenu.as
  - Level1.as

- Inherited from the Game Object Class**(Found in the GamePlay Folder)**:
  - CountdownMC.as
  - ExitGameMC.as
  - StartGameMC.as
  - ShipSP.as
  - BulletSP.as
  - EnemySP.as

Even though we will be going over all the classes during the lecture, it will be up to you to revisit them when doing the assignment in order to build the Object Manager Class **(ObjectManager.as)** and edit the GameStateManager class **(GameStateManager.as)**.

You will be given a fully structured ObjectManager class containing all empty functions that you need to implement. In order to correctly implement those functions, you need to do the following:

- Read the function's description (provided to you) and understand what the function does, what parameters are needed for it to work.
- Check how the function is used in the engine and especially in the different states, because that helps you a lot in knowing what it does.

Note: You are allowed to add private helper functions inside "ObjectManager.as" only

Comments

In this and future assignments, you are required to include:

- **A file header comment in every piece of source file. The format is shown in the "Comments.as" file given to you in the beginning of the semester and should be present at the very top of all your code.**

- **Function header for each function you create. The format is shown in the "Comments.as" file given to you in the beginning of the semester and should be present at the top of every function.**

- **Inline commenting for your code.**

PS: You should only comment the code you add in the ObjectManager and GameStateManager.
No need to comment my code.

What to submit

You must submit the **"ASEngine v2.0"** folder in a single .zip file named correctly (go to the class page on moodle and you will find the assignment submit link). **Do not submit any other files than the ones listed.**

If you've forgotten how to submit files, the details about how to submit are posted in the syllabus. Failure to follow the instructions will result in a poor score on the assignment (and possibly a zero).

Special note:

The due date/time posted is the positively latest you are allowed to submit your code. Since the assignments can easily be completed well before the deadline, you should strive to turn it in as early as possible. If you wait until the deadline, and you encounter unforeseen circumstances (like being sick, or your car breaking down, or something else), you may not have any way to submit the assignment on time. Moral: **Don't wait until the last day to do your homework.**