Vanier College
Faculty of Science & Technology
Computer Science Technology

420-203-RE Program Development in a Graphical Environment

Teacher: Sleiman Rabah

Assignment II — Fall 2022

Due date: scheduled on LÉA

Revision History

Revision	Date	Author(s)	Description
1.0	Oct. 07, 2022	S.R.	Initial handout.
1.1	Nov. 16, 2022	S.R.	Added a new constraint and a requirement.

Contents

1 Instructions and Constraints

- You must use Apache NetBeans and Open JavaFX 18.
- This assignment can be done **individually** or in a team of maximum **two students**.
- If you work in a team, both team members must **equally contribute** to this assignment.
- You are required to write your own code.
- You are not allowed to share code with, or write code for, or look at code of another/other student(s). If you do so, this will be considered plagiarism.

2 Overview

In this assignment, we will work on existing JavaFX implementation of a space invaders game. You are required to modify the provided game implementation to meet the requirements stated in the following section.

3 Implementation

Your game implementation must be based on the code that is provided along with this handout.

3.1 Requirements

The provided game implementation contains bugs and requires some improvements. You must fix the existing issues and add new features that are listed below:

1. You must search for and use your own game assets (such as sprites, explosion effects, sound files, etc.). You are not permitted to use the ones that were already included in the provided NetBeans project.
2. The spaceship's rocket is not being fired correctly.
3. Add an explosion effect when the spaceship's rocket hits a target.
4. Fix the movement of the spaceship: it doesn't move properly.
5. Generate random invader sprites using different 2D graphics.
6. Make the spaceship initially appear at the bottom middle of the scene.
7. Customize the spaceship's shield to your liking.
8. The spaceship must be controlled by the WASD keys.
9. Handle the game over case: the spaceship can be hit by invaders at most 3 times.
10. Pressing a keyboard key of your choice should change the type of rocket(s) the spaceship is firing.
11. Play a sound effect upon an explosion. Sound effect clips must be stored in the *resources/sounds/* folder.
12. Implement multiple game levels (at least 3 levels) in each of which the following requirement should be met:
    (a) The speed of the moving invaders must be increased from on level to another.
    (b) Upon starting a new level:
        i. A given number (more than 15) of different invaders (using different graphics) must be spawned. The number of invaders generated in the second and third level must be greater than the previous level respectively.
        ii. Change the sprite of the spaceship (use different one in each level).
        iii. The number of rockets fired by the spaceship must be increased.
        iv. Use different type of spaceship rockets.
    (c) Use different sound effects in each level.
13. Implement a heads-up display (HUD) in which the following items must be displayed:
    (a) The current score. The score should be updated whenever an invader gets destroyed.
    (b) The current level of the game.
    (c) The lives counter of the spaceship.

3.2 Additional Resources

1. Various game assets (such as 2D sprites, sound effects, etc.) can be downloaded from:
   - The **Open Game Art project**: https://opengameart.org/
   - Or from **Kenny**: https://www.kenney.nl/assets?q=2d
2. Make an image's background transparent.

4 Evaluation Criteria

Your work will be evaluated based on the following criteria:

Criteria	Points
Correctness of the implementation: no errors, the program works well and meets all the specifications. Ability to perform as required and producing correct output.	5%
Relevance and accuracy of the source code documentation as instructed.	2%
Code readability, naming convention, and clarity. The use of meaningful self-explanatory names for identifiers (classes, variables, constants, methods, etc.).	3%
Compliance of the implementation with the stated requirements as well as simplicity and appropriateness of your implementation.	77%
The user-friendliness of the user interface(s) and presentation of the results	10%
Overall comprehension of the submitted source code.	3%
Total	**100%**

5 What to Submit

❑ Compress (zip) your NetBeans project and upload your compressed file to LÉA **before the submission deadline**.