CS135601 Introduction to Programming (II)

Hackathon

1 Goal

In this hackathon, you are asked to finish the implementation of the partly Tower Defense game and learn the following skills:

- Understand Allegro5 game development process.
- Get familiar with the OOP concept and the code structure.
- Using Allegro5 and C++ to develop a game.

2 Problem Description

In this game, you need to control the armies to destroy all the defenses (except walls) shown on the map to win the game.

The game consists of two main components:

1. Defense: Predefine in the .txt file in the Resource/ folder

   There are currently two defenses: wall and cannon.

2. Armies: Created by the player.

   There is already an archer army in the source code as a reference.

In the playing scene:

- Press key 0-9 to change the speed multiplier.
- Press Q , W to perform a quick select on different armies.
- Press M can mute / unmute the bgm.
- Click on the empty spaces in the map to place the selected army.

The rule of armies placement:

- To simplify the game, the walls will enclose a rectangle area where the defenses are placed in.
- You cannot place the armies on the rectangle area and the walls.

However, there are some problems and incomplete parts of the code. You can solve it by following the under requirement.

3 Code Requirements (Finish the game)

1. Finish the ArmySelect scene. (0.6%)

   So far, the ArmySelect scene has only an army image with a static amount. You need to add the button to dynamically adjust the amount of each army and create a spaceCost constraint of total armies amount.

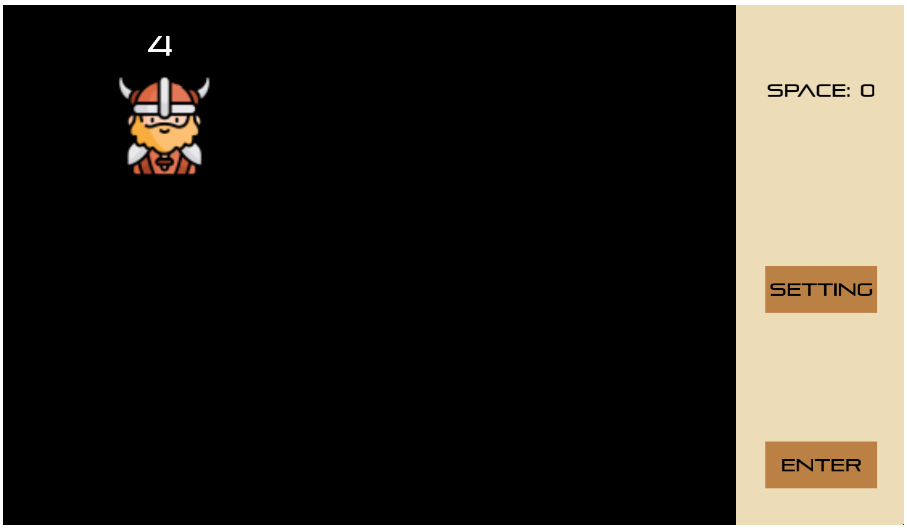
   a. Create the space usage / space limit and show it on the scene. The limit is 5.

   b. Add the add & sub button below every army image to enable dynamically modify the amount of army.

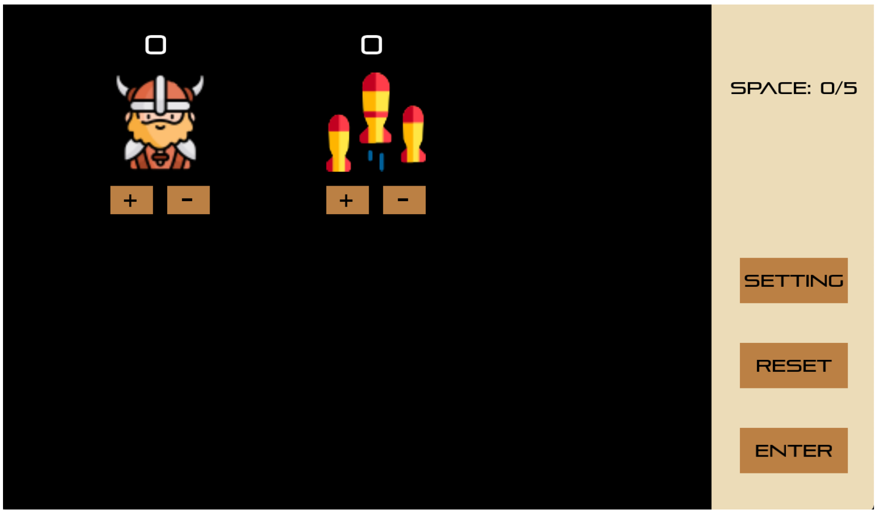
   c. Add a reset button to reset the amount of every army.

   d. The button's image should change on mouse enter/leave.

   Original ArmySelect Scene



   Complete ArmySelect Scene



2. Add 1 new army - bomb army. (0.9%)

   Currently, you cannot win the game because the walls block the movement and attack from the archer army. Hence, you need to create a bomb army, which can deal massive damage to the walls.

   a. Add the add / sub button of the bomb army in the ArmySelect scene.

   b. Add the select button below the map in the PlayScene.

   c. Bomb army prioritizes walls above all other defenses. It will destroy the nearest wall, and deal with small damage to defense when there's no wall on the map.

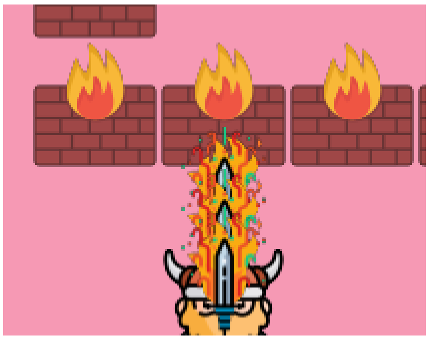
3. Add 1 new effect to archerBullet when it hits the defense. (0.3%)

   a. The effect should appear every time when an archerBullet hits the defense.

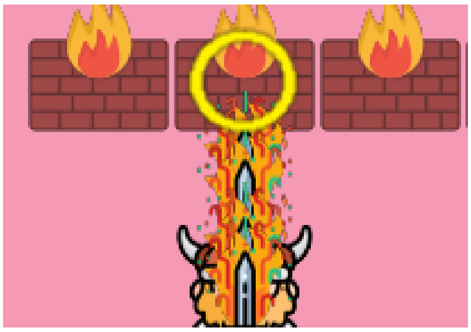
   b. The effect should appear near the defense.

   c. There are 4 images (play/shoot-1.png ~ play/shoot-4.png) that can be used.

Original



Effect



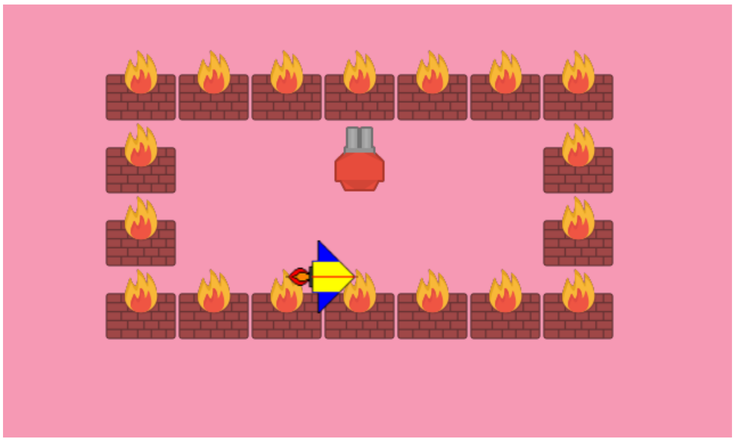
4. Set a cheat sequence to destroy all the defenses (except walls) directly (i.e. win the game) and set code for debug mode (0.6%)

    a. Set Tab as a code to active / de-active debug mode

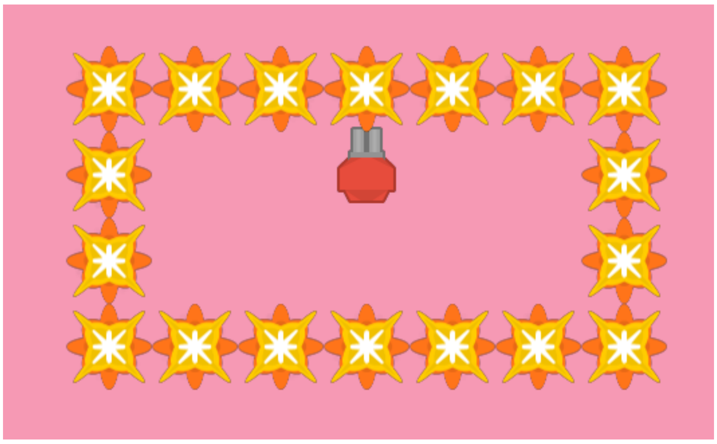
    b. Let sequence { arrow_up, arrow_up, arrow_down, _arrow_down, arrow_left, arrow_right, enter } as a code to destroy all the walls.

    If the setting is correct, you will see a plane flying through the map after the sequence.

    Plane



    Destroy all the walls



5. Fix the bugs in the game (0.6%)

    a. Normally, the game fails when all of the armies get killed while there are defenses still remaining on the map. However, the game will crash when entering the lose scene. Try to fix it.

    b. The program supposed that you can play the game repeatedly, but the archer army might move through the walls when re-entering the game. Try to fix it.

c. Strongly recommend making use of the tools in your IDE such as Stack Trace, Log, Watch variable, Breakpoint (step in / step out) to help you debug.

6. [NOTE] You can follow the TODOs in the source code to finish the above requirements.
7. [NOTE] If you finish the hackathon part on the day of the hackathon. You will get bonus points for your project 2 (1%).

4 Demo

Make sure you finish all the requirements above and demo the game to TA to get the points of Hackathon. Otherwise, you might receive a points deduction.

Since the hackathon is remote, you need to demo the hackathon by yourself. You have 5 minutes to demo the work, you need to show all the requirements above to get points of the hackathon. TAs will score only by your demonstration, so make sure you prepare before the demo. Otherwise, you might get points deduction for missing showing some of the requirements.

Demo link: hackathon demo

[NOTE] It's ok not to complete this spec on the day of the hackathon. But make sure you demo this part in the project2 demo time to get the points of the hackathon.

5 Ask Questions

In the period of QA time (i.e. 10:00-11:00, 14:30-16:00), if you have any questions, you can directly join MicroSoft Teams to ask TAs. Otherwise, please type in the chat to check TA is online first, and then join the Teams.

[NOTE] After the end of the hackathon, TAs will not answer any questions about the hackathon spec and todos. Hence, please make sure you understand everything you need to do in the hackathon, even if you aren't able to finish it.