Assignment 3 - Report

10/29/2023

10 Possible Points

Attempt 1

In Progress
NEXT UP: Submit Assignment

Add Comment

Unlimited Attempts Allowed
10/9/2023 to 11/8/2023

Details

Problem Overview

You are required to design an application model for a Space Invaders game. Space Invaders is a classic arcade game played on a rectangular screen with different types of aliens on it. The player controls a spaceship at the bottom of the screen and can shoot projectiles at the aliens. The aliens move horizontally and descend towards the player's spaceship. If the aliens reach the bottom of the screen, shoot the player, or collide with the player's spaceship, the game is lost. The player can move their spaceship horizontally to avoid the aliens' projectiles and shoot them down. The game is won when all the aliens are destroyed. The score is calculated based on the number of aliens destroyed. The duration of the game is clocked until all the aliens are destroyed.

Q: What is Space Invaders?

*A: Some real-world examples could be found **here** (https://www.youtube.com/watch?v=uGjgxwiemms) and **here** (https://en.wikipedia.org/wiki/Space_Invaders).*

Assignment 3 Requirement

In assignment 3, you are going to review and extend an existing implementation of 'Space Invader' (that is not your own) to add features, leveraging your knowledge of OOP, design principles and design patterns. Please find the detailed tasks below:

Implementation Task

What we provide to you?

- Codebase: the codebase you are going to review and extend is provided **here** (https://canvas.sydney.edu.au/courses/53385/files/33766823?wrap=1) (https://canvas.sydney.edu.au/courses/53385/files/33766823/download?download_frd=1) .

Please note that, your goal is to **extend** and **maintain** this implementation. What this means is to add features to the existing implementation by using OO design principles and appropriate design patterns you have learnt throughout this UoS, **without** breaking the implementation (it runs - rule #1 is don't break working code) or using unnecessary 'hacks'.

You are **not** required to correct the existing design of the implementation - you **must retain** the existing design wherever changes are not required and will be penalised should this be changed without cause (for example, replacing the given implementation with your own assignment 2 code). The idea here is that you work with the existing design rather than against it, and minimise required changes to the existing structure (reasonable, limited scope refactoring to support extensions is encouraged).

You are **not** required to correct the given codebase align with A2 requirements

Submit Assignment

Your Space Invaders is now expected to support the following features in the code:

- Difficulty Level
  - There are now three difficulty levels in your game including easy, normal and hard, which correspond to configuration files **config_easy.json** (https://canvas.sydney.edu.au/courses/53385/files/33766825?wrap=1) (https://canvas.sydney.edu.au/courses/53385/files/33766825/download?download_frd=1) , **config_medium.json** (https://canvas.sydney.edu.au/courses/53385/files/33766827?wrap=1) (https://canvas.sydney.edu.au/courses/53385/files/33766827/download?download_frd=1) and **config_hard.json** (https://canvas.sydney.edu.au/courses/53385/files/33766826?wrap=1) (https://canvas.sydney.edu.au/courses/53385/files/33766826/download?download_frd=1) , respectively.
    - Note: These json configuration files has alreayd been included under 'src/main/resources/' folder in your A3 Codebase.
  - The player can choose a level either by clicking on a button or by selecting from a menu or by clicking on a keyboard key (i.e., you ONLY need to implement this feature through one of these three ways).
  - You can set the easy level as the default level displayed to the player OR you can ask the player to choose a level before the start of a game.
  - **Attention Please: you are free to change the values in the sample JSON files whereas you are not allowed to change the structure of the JSON files (i.e., no added, no deleted)**.
- Time and Score
  - Duration of the game is clocked until all aliens are destroyed (i.e., game wins) or the player dies (i.e., game loses). The game must display on the screen a continually updating time (initially at 0:00).
  - The score is calculated when the player's projectile hit a alien or a alien's projectile. The game must display on the screen an updating score (initially at 0) when the player's projectile hit something during the level.
  - The alien's projectile and its corresponding score after each hit can be found in the table:

Enemy Entity	slow projectile	fast projectile	slow alien	fast alien
Score	1	2	3	4

- Undo and Cheat
    - The player can reset the game to an earlier state (including score, time, alien's position and alien projectile's position) so that a shot can be undo.
    - This undo functionality can be triggered by button, menu or keyboard action (i.e., you ONLY need to implement this feature through one of these three ways).
    - This must be a single state that is not written to disk, and the state reaching by the subsequent undo function overwrites the existing saved state (We only save one state at all time).
    - The player can do a cheating operation to remove **all aliens projectile of the same type** or **all the alien's who have the same strategy** immediately.
        - This functionality can be triggered by button, menu or keyboard action (i.e., you ONLY need to implement this feature through one of these three ways).
        - Take keyboard action as an example only: pressing the key 'p' on the keyboard will immediately remove all slow projectile on the screen at the cheating moment and add corresponding scores (i.e., 1 score X the number of slow projectile on the screen at the cheating moment).
        - Note: You need to implement both cheating options, but the player can execute only one cheating option at a time.

Report Task

Your report in this assignment must concisely cover the followings:

1. Code review on the existing codebase provided to you, which includes
    - discussion on the use of OOP or design principles (**be specific** to the given code, a UML snippet needs to be provided)
    - discussion on the use of design patterns (**be specific** to the given code, a UML snippet needs to be provided)
    - discussion on the documentation (e.g., readme file, comments, etc.)
    - discussion on how easy or difficult the given codebase and the above points made it to achieve your required functionality in this assignment and the reason
2. A discussion on your feature extension including
    - Describe the actual changes (including extensions) you have made in your code and rationalise that the changes are necessary.
    - Highlight your application of OO principles in your extensions and explain what they motivated you to do and why (**be specific** to your code, a UML snippet needs to be provided)
    - Document at least three different design patterns you have used in this assignment and rationalise their usage in terms of SOLID and GRASP (**be specific** to your code, not the pattern in general, a UML snippet needs to be provided).
        - You **must use the GoF design patterns** that we have learnt in this unit in your implementation.
        - Attention Please: you are allowed to reuse a design pattern that you have used in A2 however you are not allowed to document a repeat use case of a design pattern as A2, for example, you are not allowed to discuss applying builder design pattern to create bunker or alien, the same for factory method and strategy.
    - Reflect on your extension design, highlighting any outstanding issues or improvements or discussing your impact on the extensibility of the code
3. A UML class diagram of the after-extension version of your codebase, highlighting the design patterns you have used **for your extension** and identifying the participants in each design pattern you have used.

Submit Assignment

You are required to submit all assessment items by the due date to different portals.

You have to submit your **report on Canvas (https://canvas.sydney.edu.au/courses/53385/assignments/488235/)** AND You have to submit **your code on Edstem (https://edstem.org/au/courses/12969/lessons/43540/slides/298312)**

- Report: Submit your UML class diagram and your report as a **SINGLE** pdf document.
    - If your diagram is unreadable at 200% magnification (maximum zoom with the turnitin tool) then
        - 1) include the whole UML diagram;
        - AND
        - 2) include enlarged versions of the key components when you refer to them.
    - You must include the entire UML diagram, and you must also include enlargements of specific parts for reference in the discussion
- Code: **Submitted to Edstem by using this link (https://edstem.org/au/courses/12969/lessons/43540/slides/298312)**
    - 'gradle clean build run' will start the game.
    - A readme file named "A3_readme.txt" should cover any point you would like your marker to know
        - how to run your code (e.g., any quirks to run your application).
        - List which features (i.e. Difficulty Level, Time and Score, and Undo and Cheat) you **have implemented** in your extension.

        - List **the names of the design patterns you have used** in your extension and provide the **corresponding classnames and paticipant roles** regarding these patterns.
            - **Attention Please: class and file names that are not listed here will not be assessed as part of the design pattern.**
        - Describe how to select difficulty level, how to undo and how to cheat in your code (i.e., what operations your maker needs to do).
        - Any other info you would like your marker to know regarding your implementation.
    - We will execute your code by running 'gradle clean build run' in the terminal with the environment configuration below:
        - Gradle 7.4.2
        - JDK 17
        - Unix-based System
    - If your code fails to run using the instructions provided above, all coding parts of your submission will receive a **ZERO** mark.

Attention Please

- If you replace the given codebase with your own assignment 2 codebase in this assignment, you are going to be awarded 0 immediately for this whole assignment.

Academic integrity

While the University is aware that the vast majority of students and staff act ethically and honestly, it is opposed to and will not tolerate academic integrity breaches and will treat all allegations seriously.

Further information on academic integrity, and the resources available to all students can be found on the academic integrity pages on the current students website: **https://sydney.edu.au/students/academic-integrity.html (https://sydney.edu.au/students/academic-integrity.html)** .

We use Turnitin, which includes AI detection, to detect potential instances of plagiarism or other forms of academic integrity breach. If such matches indicate evidence of plagiarism or other forms of academic integrity breaches, your teacher is required to report your work for further investigation.

You may only use artificial intelligence and writing assistance tools in assessment tasks if you are permitted to by your unit coordinator, and if you do use them, you must also acknowledge this in your work, either in a footnote or an acknowledgement section.

Further information for on research integrity and ethics for postgraduate research students and students undertaking research-focussed coursework such as Honours and capstone research projects can be also be found on the current students website: **https://sydney.edu.au/students/research-integrity-ethics.html (https://sydney.edu.au/students/research-integrity-ethics.html)** .

Compliance statement

In submitting this work, I acknowledge I have understood the following:

- I have read and understood the University of Sydney's **Academic Integrity Policy 2022 (https://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2012/254&RendNum=0)** .
- The work is substantially my own and where any parts of this work are not my own I have indicated this by acknowledging the source of those parts of the work and enclosed any quoted text in quotation marks.
- ing, proof-reading, and automated writing and mar checkers, but not spell checkers).

Submit Assignment

- The work has not previously been submitted in part or in full for assessment in another unit unless I have been given permission by my unit of study coordinator to do so.
- The work will be submitted to similarity detection software (Turnitin) and a copy of the work will be retained in Turnitin's paper repository for future similarity checking. Note: work submitted by postgraduate research students for research purposes is not added to Turnitin's paper repository.
- Engaging in plagiarism or academic dishonesty in coursework will, if detected, lead to the University commencing proceedings under the **Academic Integrity Policy 2022. (https://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2012/254&RendNum=0)** and the **Academic Integrity Procedures 2022 (http://sydney.edu.au/policies/default.aspx?mode=glossary&word=Academic+honesty)** .
- Engaging in plagiarism or academic dishonesty in research-focussed work will lead to the University commencing proceedings under the **Research Code of Conduct 2013 (https://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2013/321&RendNum=0)** and the **Academic Integrity Procedures 2022 (http://sydney.edu.au/policies/default.aspx?mode=glossary&word=Academic+honesty)** .
- Engaging another person to complete part or all of the submitted work will, if detected, lead to the University commencing proceedings against me for potential student misconduct under the **University of Sydney (Student Discipline) Rule 2016 (http://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2017/441&RendNum=0)** .

View Rubric

Assignment 3 Report

Criteria	Ratings	Pts
Review view longer description	**3 pts** **Full Marks** A solid discussion on all four required points in the code review part. Showing a good understanding of the given codebase. Discussion is specific to the given code. **2 pts** **More than Half Marks** A discussion on all four required points but has minor flaws OR missing one required point. **1 pts** **Less than Half Marks** A discussion on all four required points but has major flaws OR Has solid discussion on only one or two required points OR the discussion is not specific to the given codebase. **0 pts** **No Marks** No discussion on the required points or unreasonable/improper discussion on all points or criticise the programmer instead of program.	/ 3 pts
Discussion -- OO principles view longer description	**1 pts** **Full Marks** A solid discussion on your application of OO principles in your extensions -- specific to your code **0.5 pts** **Half Marks** A discussion on your application of OO principles -- specific to your code -- with flaws or omission **0 pts** **No Marks** No discussion on design principles or not specific to your code	/ 1 pts
Discussion -- Design Pattern(s) view longer description	**2 pts** **Full Marks** A solid discussion on your application of three design patterns in your extension -- specific to your code **1 pts** **Half Marks** Discussion on your application of design patterns in your extension -- specific to your code -- with major flaws or omission **0 pts** **No Marks** No discussion or not specific to your code	/ 2 pts
Discussion -- Extension and Reflection view longer description	**1 pts** **Full Marks** A solid discussion on 1) actual changes/extensions you have made in your code and the reason; AND 2) any outstanding issues or improvements or your impact on the extensibility of the **0.5 pts** **Half Marks** Discussion on your actual changes/extensions and the reason AND any outstanding issues or improvements or your impact on the extensibility of the code -- with major flaws or omission **0 pts** **No Marks** No discussion on these two points	/ 1 pts
UML Diagram -- Design Pattern and Element view longer description	**3 pts** **Full Marks** Have drawn all required elements correctly AND Have correctly displayed three design pattern(s) used in your code extension and identified the participants in each design pattern correctly. **2 pts** **More than Half Marks** Have drawn all required elements correctly AND Have correctly displayed two design pattern(s) used in your code extension and identified the participants in each design pattern correctly. **1.5 pts** **Half Marks** Have only drawn some of the required elements correctly AND Have correctly displayed at least two design pattern(s) used in your code extension and identified the participants in each design pattern correctly. **1 pts** **Less than Half Marks** Have drawn the UML elements with major flaws OR Have correctly displayed only one design pattern(s) used in your code extension and identified the participants in the design pattern correctly. **0 pts** **No Marks** Haven't drawn the UML diagram OR the UML that drawn does not match your code extension	/ 3 pts
		Total Points: 0

Submit Assignment

Choose a submission type.

Upload Studio More

Choose a file to upload

File permitted: PDF

or

Canvas Files

□ I agree to the tool's **End-User License Agreement (https://api.turnitin.com/api/lti/1p0/user/static_eula)**
This assignment submission is my own, original work

Submit Assignment