CSE201 Advanced Programming (Section-A)
Instructor: Vivek Kumar
Project released on 8th November 2021

IMPORTANT Instructions:

1) It is mandatory that you attend all the deadlines in this project as per the schedule. No request for rescheduling the demo will be entertained. In case of any missed demo, we will simply carry out an offline demo without providing any online demo option any further.
   You MUST have a PRIVATE git repository for your project and every group member should frequently check in their code in this repository. Plagiarism to leave your repo public.
3) No extensions will ever be provided. Any submission after the deadline will not be evaluated. If you see any ambiguity or inconsistency in a question, please seek clarification from the teaching staff.

Plagiarism: All submitted deliverables are expected to be the result of your individual effort. You should never misrepresent someone else's work as your own. In case any plagiarism case is detected, it will be dealt with as per the new plagiarism policy of IIITD that was also discussed in the lecture.

The Will Hero Game

(Developed by ZPlay Games)

Introduction

For your AP course project, you will have to implement the game **Will Hero** using JavaFX and OOP concepts. You can find a youtube video of this game at this link. This game is also available on Android Play Store and Apple App Store. However, you must use the game version that appears on youtube, as mentioned above, for your project implementation. We have made several changes to the game as described below in detail to simplify your project implementation. Note that if you download the game on your phone, that version could be very different from the one shown in the youtube link. Hence, while you are encouraged to play the original game on your phone, you must follow the youtube link and game descriptions provided in this document for implementing your project. You must also ensure that you read all the instructions provided in this document before posting any query on google classroom.

Game Summary

The Hero hops between floating islands where he encounters enemies known as Orcs and some obstacles. The objective of your project is to make the Hero manoeuvre through these islands and defeat the Boss. Boss is also an Orc, although with more power and of a bigger size. The Hero can kill Orcs either by pushing them into the abyss or by killing them using some weapons. While manoeuvring, the Hero can find treasure chests that could give him some weapon or some coins that the Hero has to collect. The player loses the game if the Hero falls into the abyss or fails to defeat the Boss.

Gaming Rules

1. There are several possible worlds (background) in the original game that the player can choose as the game background before starting the game. Some of these worlds are Jotunheim, Ghostville, Tropics, Downhill and underworlds. The original game also has several levels, where each level is concluded when the Hero frees the Companion from the Boss. However, in your implementation, you can only support the default world as the background and only the first level of the game. (See Video at 0:28 to 0:35)
2. The Hero always keeps jumping at its current position. Each of these jumps is of fixed height. The player has to use a single input to move the Hero once in the forward direction. The Hero can only move in the forward direction of a fixed length with each input. The game should always display the Hero's location at the top. Location at any point is simply the total number of forward movements of the Hero. Your game could have a total of 122 locations, where the Boss fights between locations 107 to 112. (See Video at 7:01 to 7:12, and the location is displayed at the top, e.g., the number 107 at the duration 7:01). The number of moves is not fixed, and you are allowed to add extra moves for a smoother transition.
3. The original game supports various kinds of Orcs, as mentioned in this link. Apart from the Boss, your implementation must have at least two different Orcs (choose one from red and one from green Orcs). The Hero can defeat each Orc either by pushing it into the abyss or by killing it using some weapon. Refer to the video for how Hero can eliminate different Orcs.

4. While manoeuvring, the Hero in the original game can encounter different kinds of obstacles as well, such as TNT (See Video at 2:55), falling platforms (See Video at 4:20) and Windmills (See Video at 2:30). However, you can choose either one of these obstacles only for your implementation. In the original game, the Boss stands on a falling platform, but you can keep it on a fixed platform.
5. The Hero can hop between floating islands only. Failing to land on an island will make the Hero fall into the abyss (See Video at 1:37), ending the game. The Hero can also be eliminated by an obstacle or by an Orc. In all such cases, you should provide an option to resurrect the Hero in exchange for some fixed number of coins. Only one resurrection is allowed per game, and you can choose the number of coins required for resurrection.
6. The Hero can be equipped with a fixed helmet at the start of the game. The original game has different types of Helmets. Each Helmet comes with a set of four Weapons. You can choose to support any one helmet and any two types of weapons only in your implementation. You are free to choose the two weapons from the list provided in the above link. Whenever the Hero kills an Orc, he should be rewarded with some number of coins. You are free to choose the number of coins.
7. The Hero will find Chests placed throughout the game level. You can only support Coin Chest and Weapon Chest. The Coin Chest rewards the Hero with some number of coins that the Hero collects. Your chosen Helmet will support two different types of weapons, but the Hero doesn't have any weapons available with him initially. He will be able to get these weapons only if he can find Weapon Chests. Each Weapon Chest can equip the Hero with only one type of Weapon (See Video at 0:42). If the Hero already has that same Weapon, the weapon is upgraded (See Video at 6:40). You are free to choose any two Weapons for any Helmet you decide to implement.
8. The Hero wins the game if he defeats the Boss at the end of the level (See Video at 7:08 to 7:15). After defeating the Boss, the Hero releases a pink companion in the original game, but this is not mandatory for your implementation. You can declare the Hero as a winner once he can defeat the Boss.
9. Your implementation must allow the player to start a new game, save the current game or load a previously saved game from any point during the gameplay. For this, you must save the Hero's current position, Helmet with weapons unlocked, and the coins collected so far. Your game should allow multiple saves and reloads.
10. As mentioned above, you are free to choose how many coins each chest will give, the coins awarded on killing Orcs, and coins required to resurrect the Hero.

Notes / Hints

It is not mandatory to have a fluid-like animation as shown in the game. You must attend the UML tutorials (09/11 and 11/11) and JavaFX tutorial (16/11) to learn more about how to approach your AP project deadlines. We will cover object serialization / deserialization in the lecture on November 16th. These concepts are required for saving/reloading the game.

You will observe that many images come up during the gameplay. You are not required to design all these images yourself, and instead, you can download these images from the internet. You can refer to this wiki page for further information and images. However, we do not

guarantee the correctness or completeness of this page. Minor changes in visualization (GUI) and animation are allowed, as long as the above-mentioned gameplay rules are not violated.

Remember that some game functionalities have been eliminated, such as stars, trophies, the concept of the tower & special abilities, Companions etc., that can be found while playing the original game to simplify your implementation.

Bonus Marks

Although we have specified the basic requirements, if you can come up with some more exciting features, you would be "eligible" for bonus marks. Although this is eligibility, we will decide based on factors such as how many other groups have also come up with the same additional functionality and how significant is that functionality.

NOTE: simply implementing the original features removed for this project, as described above in the Gaming Rules, will not be considered for the bonus. **You must do something interesting/insightful and use your creativity to earn a bonus.** There will be a single bonus for any number of bonus points you choose to implement. Exact bonus marks will be announced after the Deadline-3 is over.

Project Deadlines:

Deadline	Due Date	Deliverables
Deadline 1	21st November	Submit detailed UML class diagrams and use case diagrams for your project. Submit pictures of your diagrams on google classroom. We will use these pictures during the demo of this deadline.
Deadline 2	5th December	Show static GUI of your project and also some animation components. Submit this code on google classroom just like assignment deadlines.
Deadline 3	24th December	Submit complete project on google classroom.