German University in Cairo
Faculty of Media Engineering and Technology
Menna ElShahawy
Mohamed Mohy
Nada Nasser
Reham Ayman
Youstina Melek

Advanced Computer Lab Winter 2023
Mini-Project 1: Go Go Power Rangers
Due: 11:59 pm October 25th 2023

---

1 Project Objective

The main aim of this mini-project is to test your ability to develop the core parts of a video game that can be deployed on both desktop and mobile (*bonus*) platforms. In this project, you will be creating a simple game based on the popular Power Rangers series. The game is an endless/infinite runner [(1)(2)] which is similar to ***Subway Surfers***. The following links might be helpful to you during your implementation:

- https://docs.unity3d.com/Manual/InstantiatingPrefabs
- https://docs.unity3d.com/Manual/setupmultiplescenes
- https://docs.unity3d.com/Manual/PublishingBuilds
- https://learn.unity.com/tutorial/introduction-to-object-pooling
- https://docs.unity3d.com/Manual/android
- https://learn.unity.com/tutorial/touch-input-for-mobile-scripting-1
- https://freesound.org
- https://www.purple-planet.com

2 Gameplay Overview

The player controls a sphere that moves forward automatically on an infinite floor. The floor is divided into 3 lanes. The player must avoid obstacles while collecting different orbs to switch to different forms and utilize different powers. The player encounters both orbs and obstacles along the road. Orbs increase the player's resources, while touching blocks reverts the player's form and eventually ends the game. The player's goal is to achieve the highest score before the game ends.

[(1)] https://en.wikipedia.org/wiki/Category:Endless_runner_games
[(2)] https://www.androidauthority.com/best-endless-runner-games-android-690566/

3 Rules of Play

3.1 Movement

- The player can view the game from the third-person perspective (i.e. behind) of the sphere.
- The player automatically (i.e. not controlled by the input) moves forward (think about the final effect and not just the literal meaning of the sentence).
- The player can steer left and right to change lanes. Steering can either be continuous or discrete.
- The player cannot exit the playing area (i.e. 3 lanes) while steering right/left.
- The player's speed (i.e. forward and sideways) should be not too fast or too slow and should be consistent across devices (think about frame rate).

3.2 Item Generation

- Game items (i.e. orbs and obstacles) are generated automatically and randomly across 3 lanes throughout the entire game.
- The player encounters **red, green and blue orbs** that can be collected during gameplay.
- The player encounters **obstacles**, each fully occupying 1 lane, that should be avoided during gameplay.
- Each lane can either be empty or occupied by exactly 1 game item generated in its center.
- On the same horizontal line, there can exist 1, 2 or 3 game items.
- On the same horizontal line, there can exist a maximum of 2 obstacles.
- On the same horizontal line, all possible permutations with repetitions of game items **must** be generated, except consisting of 3 obstacles.
- There should be multiple horizontal lines of generated items visible in front of the player at all times.
- There should be enough distance between horizontal lines for the player to maneuver.
- All game objects should be either pooled or discarded after they are no longer needed (i.e. they go past the player). This is done in order to avoid memory leaks, which could lead to crashes.

3.3 Energy & Score

- The player has 3 different types of energy points: red energy points, green energy points and blue energy points.
- The player's energy points' are initially set to **zero**, and their maximum value is **5 points**, and can never exceed this value.
- The player has a score counter, which is initially set to zero.
- Collecting an orb normally increases **both** the **score** and the corresponding colour's **energy points** by 1 point each.

3.4 Switching Forms

- The player initially starts in **normal** form (i.e. white).
- Only when the energy points of a particular colour are at their **maximum** value (i.e. 5 points) can the player switch to its corresponding form by pressing "J", "K" and "L" to switch to red, green and blue forms respectively.
- Switching to a form changes the player's colour and consumes **1 point** from its corresponding energy points.
- The player can switch to any form from either the normal form or a different colour form, as long as the energy points of said colour are at their maximum. If the player attempts to switch to a form without having enough energy points, they remain in their current form.
- When the player collects an orb of the **same** colour as their current form, the **score** is increased by **2 points**, while the **energy points** are **unaffected**. However, collecting orbs of other colours behaves normally.
- When in an active form, and its corresponding energy points reach **zero**, the player automatically reverts to the normal form.

3.5 Powers

- When in a form, the player can **only** use the power corresponding to that particular form by pressing "Space bar", which consumes **1 point** from its corresponding energy points.

3.5.1 Red Power (Nuke)

- Once activated, all currently **existing obstacles** ahead of the player are **eliminated**.
- After all the existing obstacles are eliminated, obstacle generation should continue normally.

3.5.2 Green Power (Multiplier)

- Once activated, the **next orb** to be collected provides **5 times the score points** and **double the energy points**.
- When collecting orbs with the **same** colour as the current form, the **score increases by 10 points**, however, the **energy points** do **not** increase.
- When collecting orbs with **different** colours from the current form, the **score** increases by **5 points** and the **energy points** by **2 points**.
- Multiplier is automatically deactivated upon collecting **only** 1 orb.
- Pressing "Space Bar" while the multiplier is active has **no effect**. The player does not gain an extra multiplier, and no energy points are consumed.
- Switching to another form, either inadvertently by hitting an obstacle or by pressing a switch form button while having enough energy points, removes (i.e. deactivates) the multiplier automatically.

3.5.3 Blue power (Shield)

- Once activated, a shield surrounds the player which **protects** from **1 obstacle** hit.
- The shield is automatically destroyed (i.e. deactivated) upon hitting **only** 1 obstacle.
- Pressing "Space Bar" while the shield is active has **no effect**. The player does not gain an extra shield, and no energy points are consumed.
- Switching to another form by pressing a switch form button while having enough energy points, removes (i.e. deactivates) the shield.
- The shield is **not** affected by collecting orbs.

3.6 Obstacle Damage

- The player reverts back to normal form upon hitting an obstacle in coloured form.
- The player cannot be damaged multiple times by the same obstacle.
- The player dies upon hitting an obstacle when in normal form. In this case, the game ends and a "Game Over" screen is displayed.

4 Game Controls

4.1 Windows

- The player moves left and right using both the left and right arrows and "A" and "D".
- The player switches to different forms by pressing 'J", "K", and "L" to switch to red, green and blue respectively.
- The player activates the current form's power by pressing "Space Bar".
- The player pauses and resumes playing by pressing "Escape".

4.2 Android (*5% Bonus*)

The game should be only playable in **landscape** (i.e. horizontal) mode only.

- The player moves left and right by swiping the screen to the left and right.
- The player should be able to switch to any of the forms by pressing the corresponding form button. There should be a button for each of the 3 forms, stacked **vertically**. These buttons should be located at the **left side** of the screen.
- The player should be able to activate their power by pressing a power button on the **right side** of the screen.
- The player pauses by pressing a pause button on the **top** of the screen.

Note: The buttons for the Android controls do not have to be hidden in the Windows version.

5 Screens & UI

5.1 Title Screen

1. Play
2. Options
   (a) How to Play: A brief description of the rules of play and the buttons needed to play the game.
   (b) Credits: Any audio sources, materials, or external resources used **must** be credited.
   (c) Mute Sound: A **toggle** button that can mute/unmute all sounds (i.e. soundtracks and sound effects) within the game.
3. Quit

5.2 Gameplay HUD (Numerical Points)

1. Red Energy Points
2. Green Energy Points
3. Blue Energy Points
4. Score Counter

5.3 Pause

1. Resume
2. Restart
3. Main Menu

5.4 Game Over

1. Final Score
2. Restart
3. Main Menu

6 Sounds

6.1 Sound Effects/Feedback

- Whenever the player collects any of the orbs.
- Whenever the player switches between forms.
- Whenever the player uses any of the powers.
- Whenever the player hits any of the obstacles.
- Whenever the player makes an invalid action (e.g. form switch or power usage with insufficient energy).

6.2 Soundtracks

- Slow-paced track for the title, pause, and game over screens.
- Exciting and/or tensing soundtrack for the game.

7 Cheats - Optional

Implementing cheats will help us to test individual aspects of your project, just in case we were not able to test it throughout the game. Cheats **only** need to be available on the **Windows** version.

Suggested Cheats

- Pressing "U" toggles the player's invincibility. In this state, obstacles do not affect the player. However, both the energy points and score behave normally.
- Pressing "I", "O" or "P" to increase the player's red, green and blue energy points by 1 point respectively.

8 Guidelines

- This mini-project should be done **individually**. You can share ideas, consult the manual, and search online. However, all work done in this mini-project must be done by your hands and your hands only. This means that copying code from online resources or ChatGPT is not allowed.
- You are **not required** to use external assets. However, you can include external assets (e.g. audio, music, 3D models, etc.) if and only if **any and all of them are credited in the credits section of the title screen**.
- Use **Unity** and **C#** to create this project. Once you are done, build the project for **Standalone Windows** and compress it into a .zip file (not .rar). Make sure to include **all** files generated on build e.g. **"_Data"** folder, **"UnityPlayer.dll"** file, **"_.exe" file, etc**.
- For plagiarism checks you must provide the **"Assets" folder** of your project. This can either be provided as a link to **a private Git repository**, or **a .zip file on your Google Drive which is accessible by "Anyone with Link"**.
- If you opt for the Android bonus, you must also build and provide the "_.apk" file.
- The above files: .zip .apk (if exists), should be named in the following format **"Tutorial_ID_Name"**, for example: **"T01_49_1234_Adam.zip"**.
- The android bonus is worth **5% of the project grade**, which is equivalent to **1% of the course grade**. In order to receive the bonus, **all** android requirements should be satisfied. No partial grade will be given if **any** requirements are missing.
- Use the following link to submit your Mini-Project 1 files: https://forms.gle/49cwbSNs1CLjzG2P9