



WIX1002 FUNDAMENTALS OF PROGRAMMING
MANAGERIAL REPORT

ATTACK ON TITAN



LECTURER NAME : DR. ANG TAN FONG

COURSE OCCURRENCE : OCCURRENCE 1

GROUP NO. : GROUP 6

STUDENT NAME	MATRIC NO.
LIM BOON GUAN	U2005273
FRANKIE LIM QI QUAN	U2005263
GOH CHEE LAM	U2005382
TAN THAI MING	U2005417
DARREN SOW ZHU JIAN	U2005252
TAN WEI JIE	U2005400

Project Description

This project is called “**Attack on Titan - Parallel World**”. It is a basic turn-based offline tower defense game running on the command-line interface (CLI). Players will need to help humanity to counter titan by building weapons on walls. Titan will try to destroy the wall and this action must be prevented in order to protect residents within the wall. Players will need to collect coins over time after starting the game and the collected coins will be used for buying weapons and upgrading weapons and walls. Weapons should be able to kill the titans before they destroy the wall in order for players to keep surviving the game and eventually to win the game. Otherwise, the weapons on the walls will be destroyed by titans and any walls HP drops to 0 will result in game over.

Project Requirements

In this project, the minimum requirements are to create a grid-based game interface as the battleground, 10 units of walls with hit points, coins as the only game resource that are generated over time, weapons to be built on the wall as defensive measures to attack the titans as well as titans as enemies. To add, there are two types of titan which included colossus titan and armoured titan. Each with a very attacking method respectively. However, we also add some additional features to spice up the gaming experience which are background music during game play, shield for random titan, basic background color instead of regular black and white, endless mode and hard mode in addition to normal gameplay mode for challenge seeker as well as a leaderboard for every mode of gameplay.

Approaches Taken

We use Java language to code our project. Thus, we opt Apache NetBeans which is our Integrated Development Environment (IDE) for Java as our development tool. However, other IDE is acceptable as long as we know how to use it. For example, IntelliJ IDEA, Visual Studio Code, etc. Initially, we did some discussion and research about basic game development and approaches to meet the

requirement of the project via various platforms and finally was able to come out with an effective method.

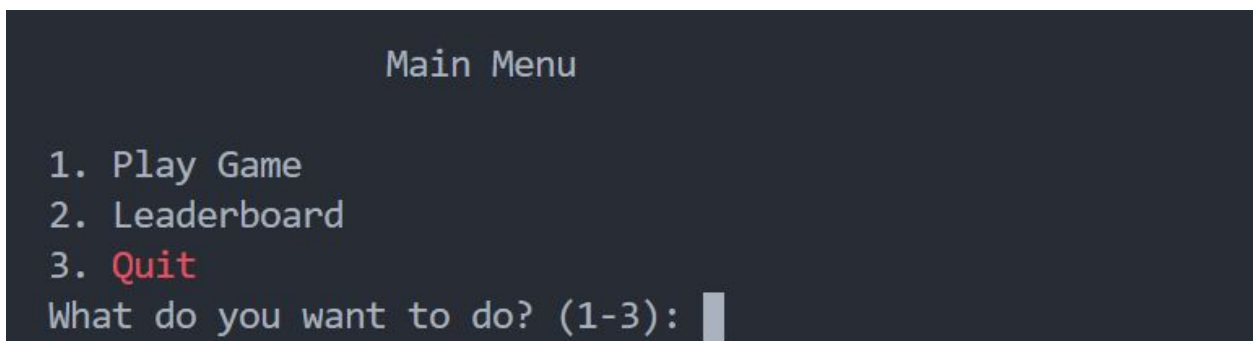
We all agreed on using GitHub as the collaboration tool. Github supports many great features. Using GitHub, we can see what changes have been made in our codes. We can easily clone the repositories to our local computers and start our projects immediately. After updating the existing code, we can push the code back to the GitHub so other group members are able to see it. Notifications are also sent to each group member as a reminder.

Besides that, we also watched ‘Attack on Titan’ which is a popular Japanese anime to obtain some infos about its worldview. This can help us to design the welcome screen and banner of our game better. We decided to use the piano version of the anime theme songs as the background music of the game. We downloaded it from Youtube which is a popular online video-sharing platform.

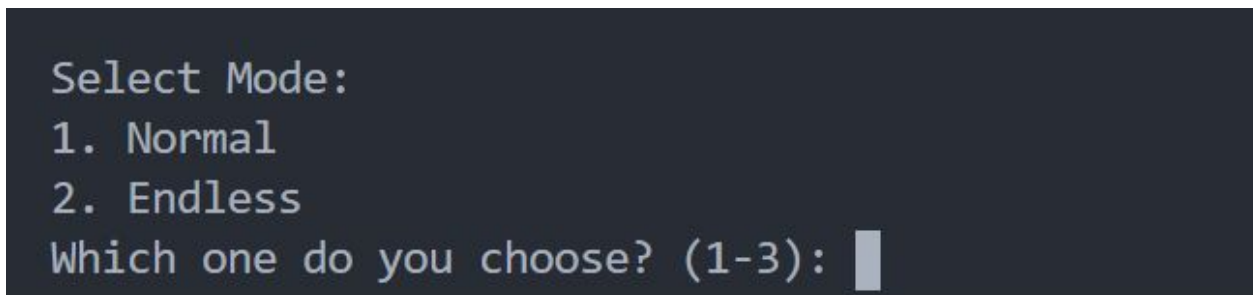
Before the coding process, we have a discussion among our group members using video conferencing platforms like Microsoft Teams and Google Meet. After that , we write pseudocodes and draw flowcharts of each Java class required to represent the algorithm of the whole project.

Main Process

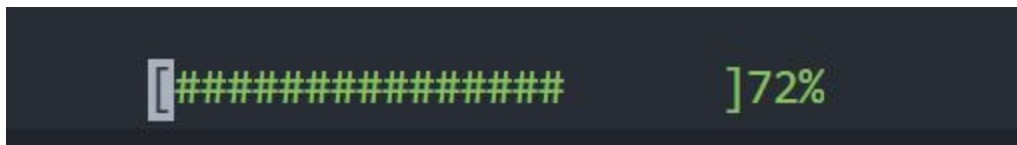
When running the game, background music will start playing. Game will then get user input as the player's name. After that, the user will need to enter numerical input (1, 2, 3) to either play, view leaderboard or exit the game.



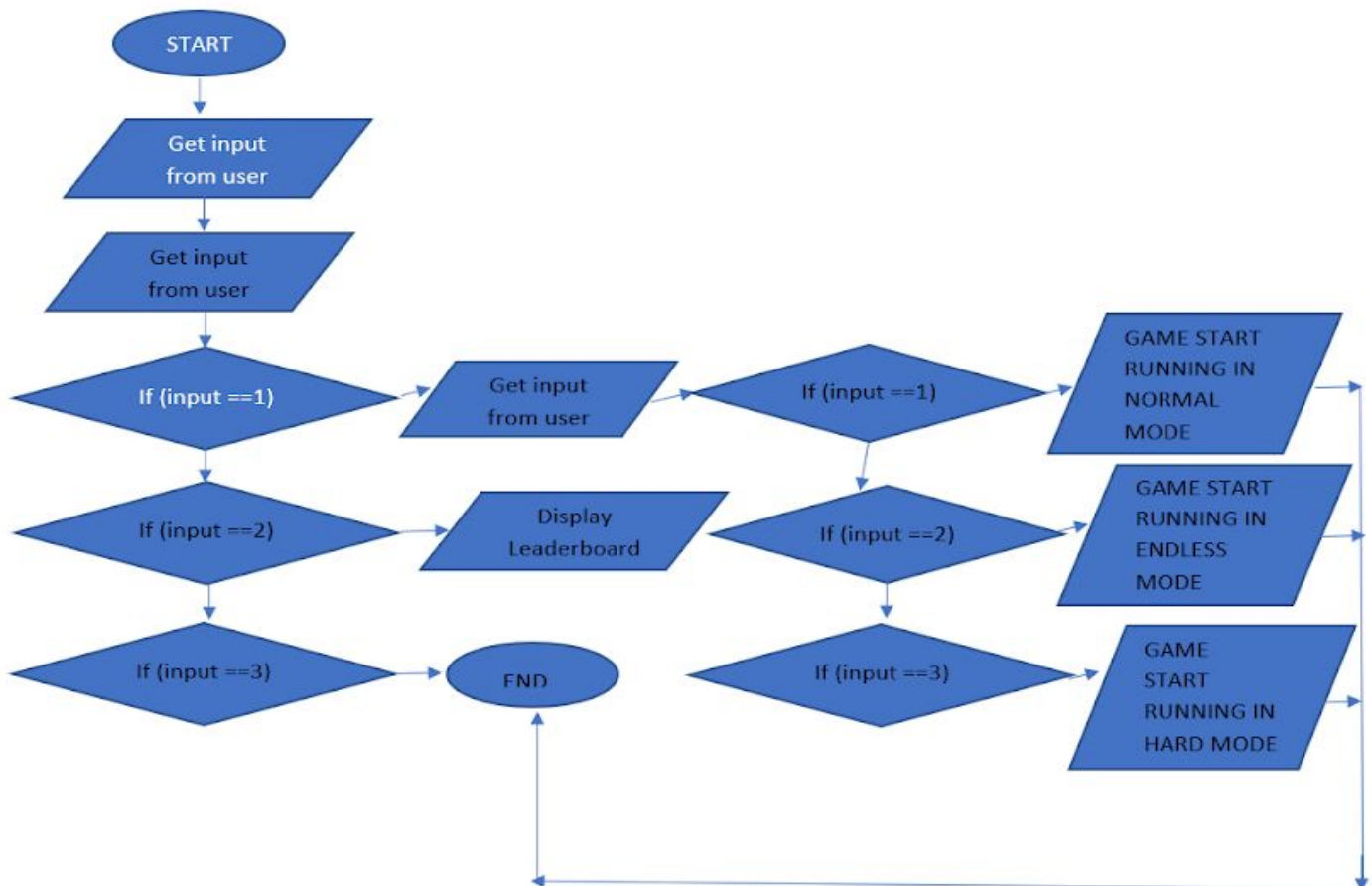
If the user opts to play, the game will then require the user to enter another numerical input to choose the mode of game which includes normal and endless.



Then, the game will begin soon after the loading bar reaches 100%.



Flowchart:



Ground.java

We create a Ground class as the battleground of our game. The dimension of the Ground is 10 units x 10 units which consists of Row 0 to 9 and also Column 0 to 9. The titans are placed on Ground. The Armoured Titan is generated on Row 0 while the Colossus Titan are placed along Row 9. Each coordinate on Ground (Row x Column) consists of three spaces. Two spaces are allocated to the overlapping of the titans and one space is for the generation of the shield on that particular coordinate.

```
Player's turn
On The Ground
Row
0
1
2
3
4
5
6
7
8
9
  A
  C
  ---
  0  1  2  3  4  5  6  7  8  9
```

Wall.java

The wall class represents each wall unit in our game. We use a constructor to set initial wall hit points(HP) to 50. A toString() method will return a string to be displayed as the wall patterns on the screen which consists of the infos like HP, index and the edge of the wall as shown below.

---	---	---	---	---	---	---	---	---	---		The Wall
0	1	2	3	4	5	6	7	8	9		Index
50	50	50	50	50	50	50	50	50	50		HP
---	---	---	---	---	---	---	---	---	---		

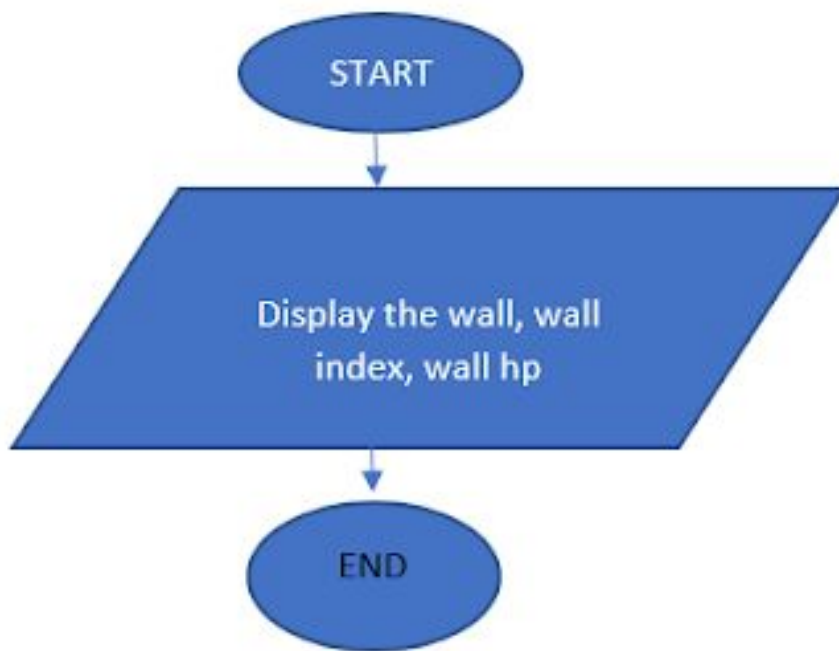
A wall can be upgraded by paying coins to add the HP of the wall. The amount of HP added is equivalent to the numbers of coins paid (i.e. 1 coin = 1 HP added to the wall).

```
Do you want to upgrade all walls? (press 1 if yes, press Enter if no) Current coin number: 50
1
How many HP do you want to add up to all the walls? Current coin number: 50
5
Total Cost: 50
```

If any wall units HP drops below 0, it is destroyed by titans and the player immediately loses the game.



0072 07



Coin.java

Coin is the only resource of the game in order to upgrade walls, buy and upgrade weapons. Coin class was created to perform those actions and keep track of current coin balance. The number of coins given to the player as the start of the games is 50.

```
HOUR 0  
Coin: 50
```

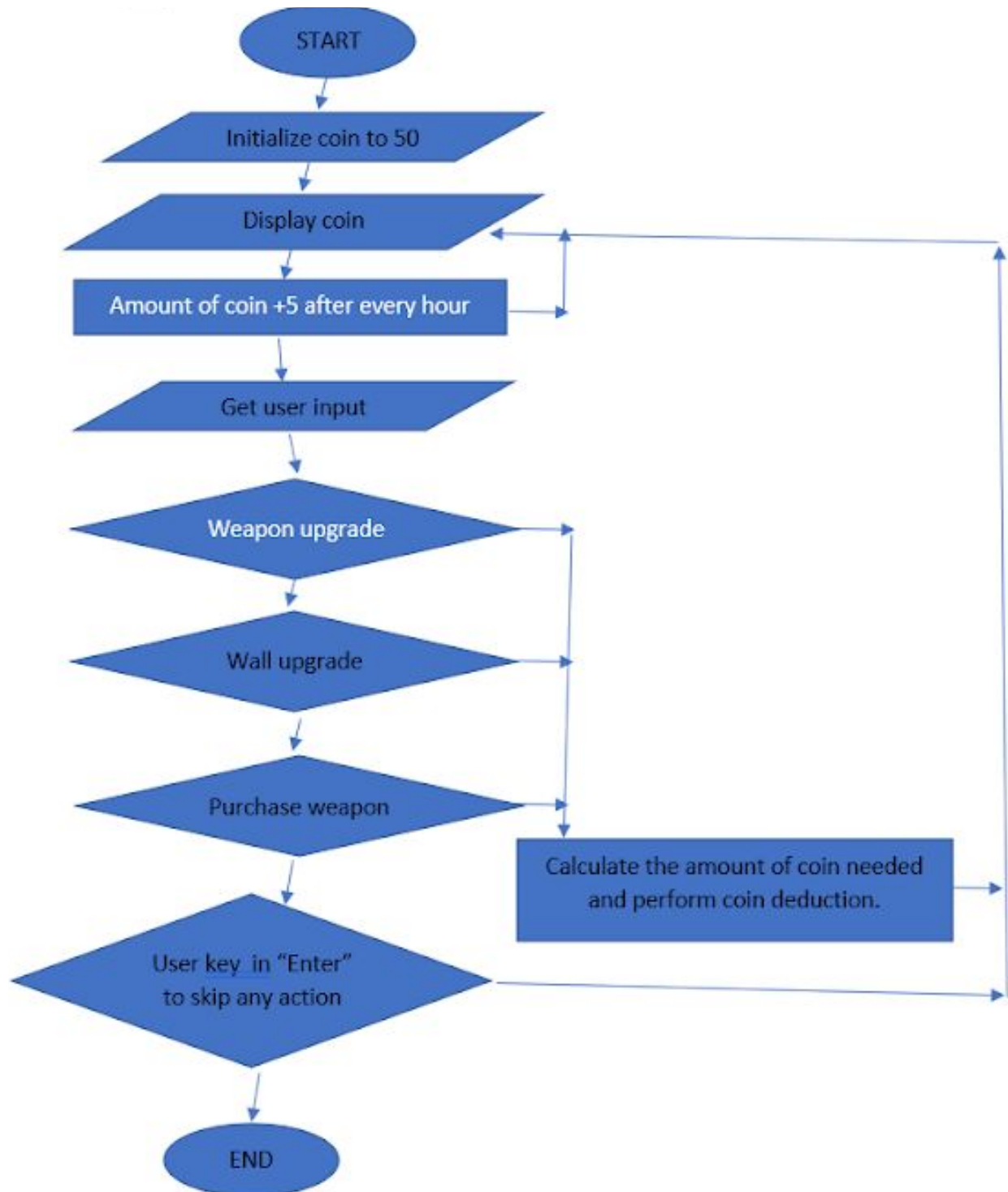
In addition, the coin will generate over time at the rate of +5 coins per hour.

```
Coin + 5
```

```
HOUR 1  
Coin: 55
```

Coin class has a pay() method to deduct the upgrade cost from the coin balance as long as the upgrade is valid which means that the upgrade cost is always smaller than the current coin balance. If the coin is not enough to upgrade a weapon or wall, a message “Not enough coin!” is displayed.

```
How many HP do you want to add up to the wall(s)? Current coin number: 0  
2  
Not enough coin!
```



Weapon.java

Weapon class was created to represent the weapons on the wall. It has instance variables like level and damage. A level 0 indicates that the weapon is not built on a wall or it is damaged by titans. A weapon can be upgraded to the next level by paying coin which equals to next level damage (i.e. Upgrading the weapon from level 1 to level 2 requires 5 coins). The toString() method is used to display the pattern of weapons on the wall units

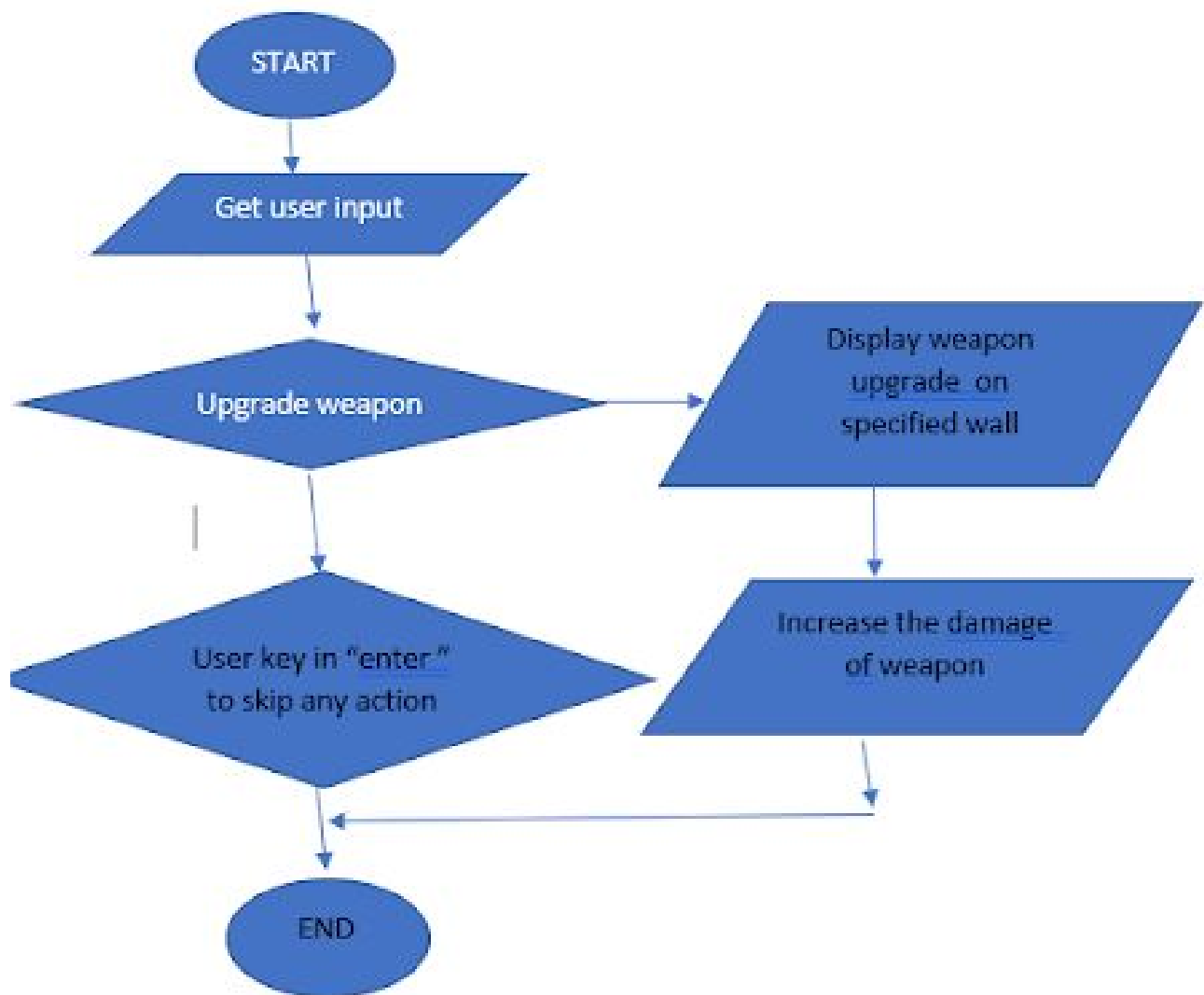
```
          ***
        *** ***
      *** *** ***
-----
 0   1   2   3   4   5   6   7   8   9   The Wall
50  50  50  50  50  50  50  50  50  50  Index
-----
HP
```

The player can upgrade the weapons by specifying the wall index which the weapon is located on in a string of integers (i.e. “123” indicates player wants to upgrade weapon on wall with index 1, 2 and 3). The upgrading of weapons is performed by adding levels of the weapons and increasing the damage done to the titans. (User can press “Enter” key during gameplay to skip any action)

```
Choose the weapon(s) you would like to upgrade (Type a string of integer or hit Enter to skip)
0123
Total Cost: 8
```

```
*** *** *** ***
-----
 0   1   2   3   4   5   6   7   8   9   The Wall
50  50  50  50  50  50  50  50  50  50  Index
-----
HP
```

The weapon on the wall can be attacked by Armoured Titan, resulting in the decrease of weapon level by 1. We need to build more high level weapons to survive the attack from titans.

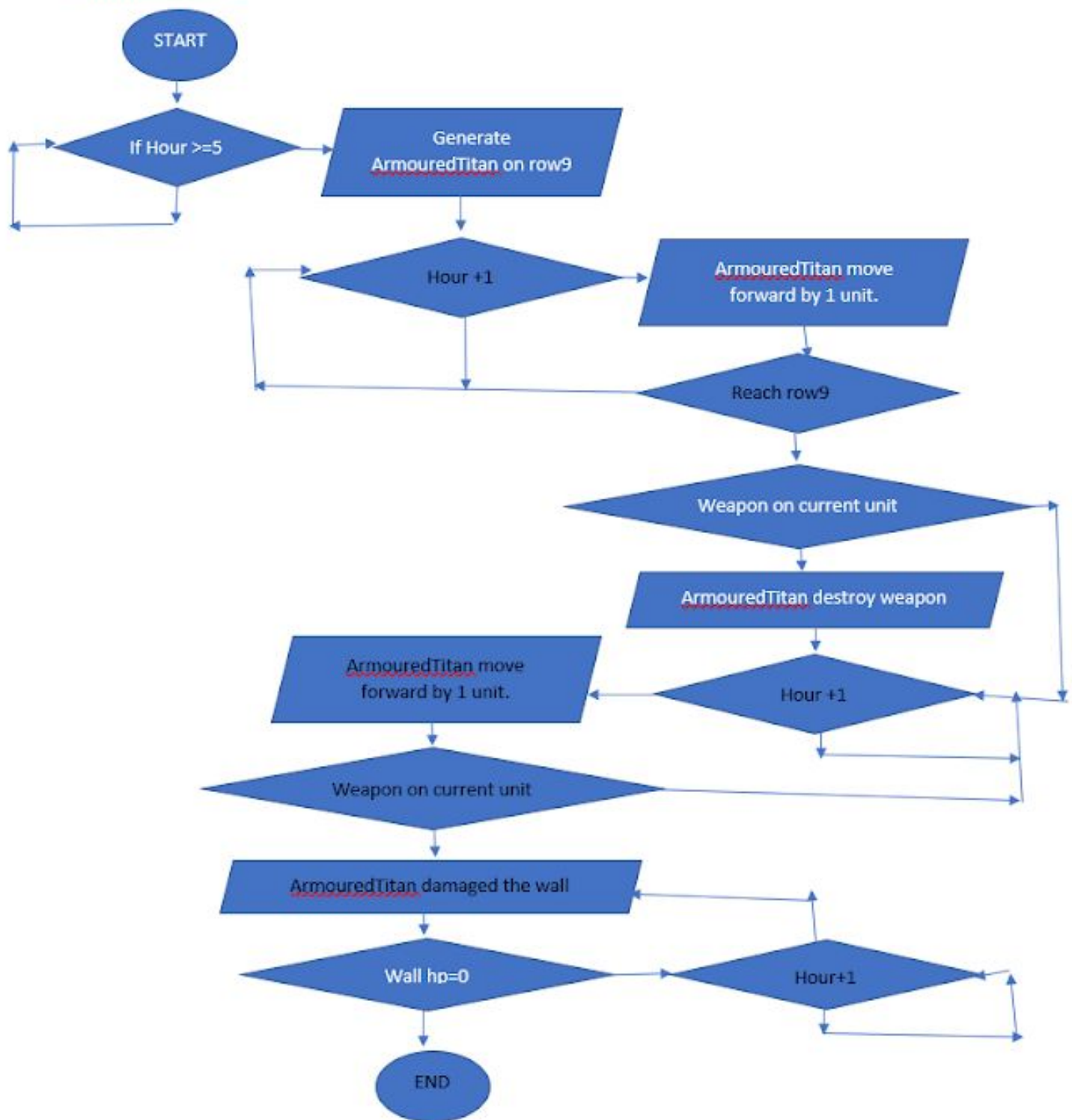


Titans

There are two different types of titans in this game which are Colossus Titan and Armoured Titan. Titans will take damage when there is a weapon on the same lane (column). Thus, two classes were created namely ArmouredTitan and ColossusTitan.

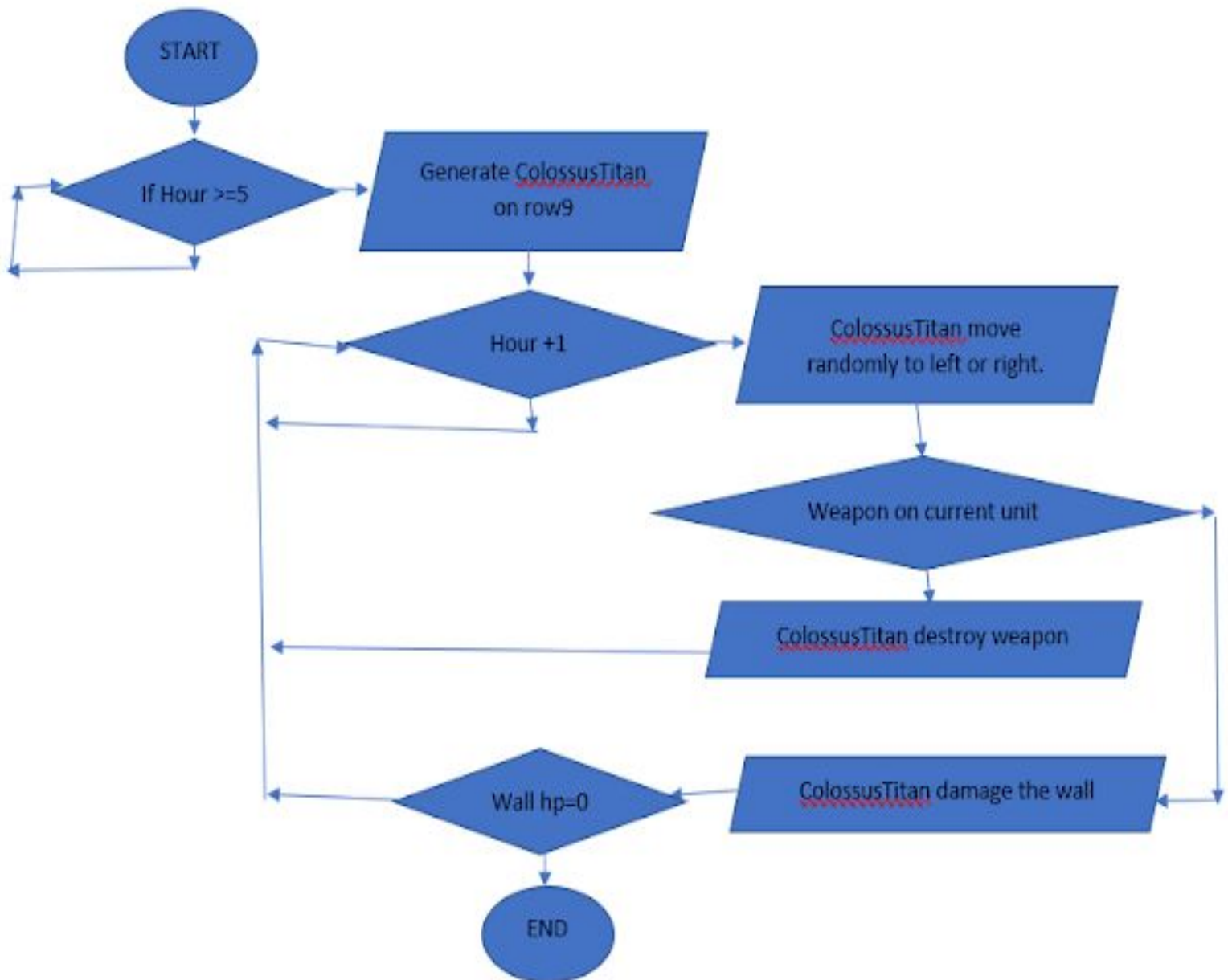
❖ ArmouredTitan.java

Armoured Titan will only start to be generated after Hour 5 and it will only show up on Row 0. Each ArmouredTitan will have 100 hit points (HP) and 5 points of attack. After every hour, the Armoured Titan will move closer to the wall with weapons by one column at a time. Upon reaching Row 9, it will attack the weapon on its lane. The level of the weapon attacked by Armoured Titan will be reduced by 1 (i.e. Level 2 to Level 1). If there is no weapon on the same column with Armoured Titan, it will move closer by one column to another wall with the weapon on it. When there is no weapon on the wall, it will start to damage the wall unit on its lane.



❖ ColossusTitan.java

Colossus Titan will show up on Row 9 after Hour 5. Each Colossus Titan has hit points of 50 and 10 points of attack. It will only move to left or right randomly. It does not destroy weapons in the same column with it. However, it will deal 10 damage to the wall units.



Each coordinate on Ground can hold 2 titans at the same time. The overlapping of Armoured Titan and Colossus Titan is represented as 'AC'. The overlapping titans will take the same damage from the weapon (AOE).

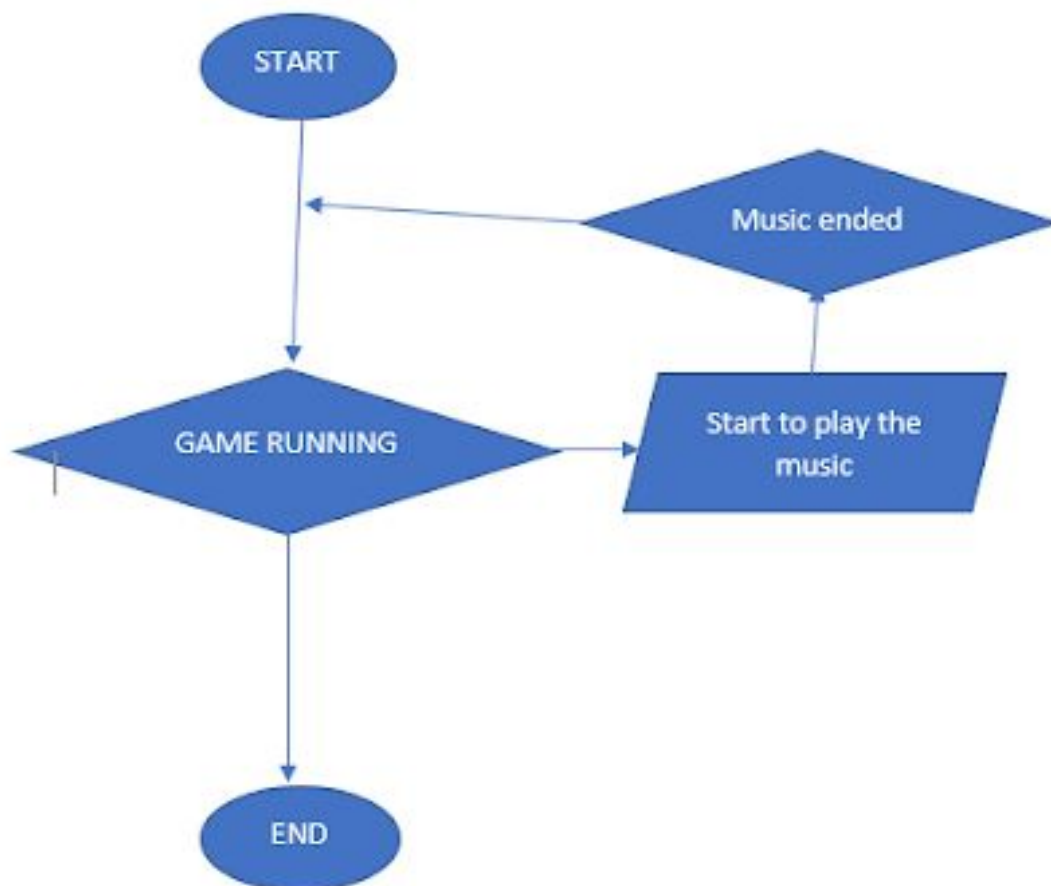
```

Player's turn
On The Ground
Row
0
1
2
3
4
5
6
7
8
9
    A
    C
    The Wall
    Index
    HP
    0    1    2    3    4    5    6    7    8    9
    50   50   50   50   50   50   40   50   50   50

```


Music.java

Music class was created to play the background music. The javax.sound.sampled package was imported for this purpose. The music will continuously be looped until the game is over.



SFX.java

We created a SFX class which is almost similar to the Music.java but it offers different functionalities. The SFX class will handle the sound effects like screaming of titans, pumping of gas and a lot more for you to discover in the game. It also makes sure that the subtitles are synchronized with the audio of the trailers when we enter the game.

```
その時  
At that moment  
  
思い出した  
I realised  
  
そうだ  
  
この世界は  
that this world
```

```
Counting the coins...  
OK  
  
Writing to leaderboards...  
OK  
  
Preparing the arena...  
OK  
  
Pumping the gas...  
OK  
  
Testing the weapons...  
OK
```

!!!!!!!

か。か。。。壁に穴穿けられた

They blew a h-hole into the w-w-wall.....

いってくるぞ!

They're coming!

巨人たちが入ってきた!!!!

THE TITANS ARE COMING!!!

ダメなんだ。。。.

It's all over.

この町を

This town is...

もう

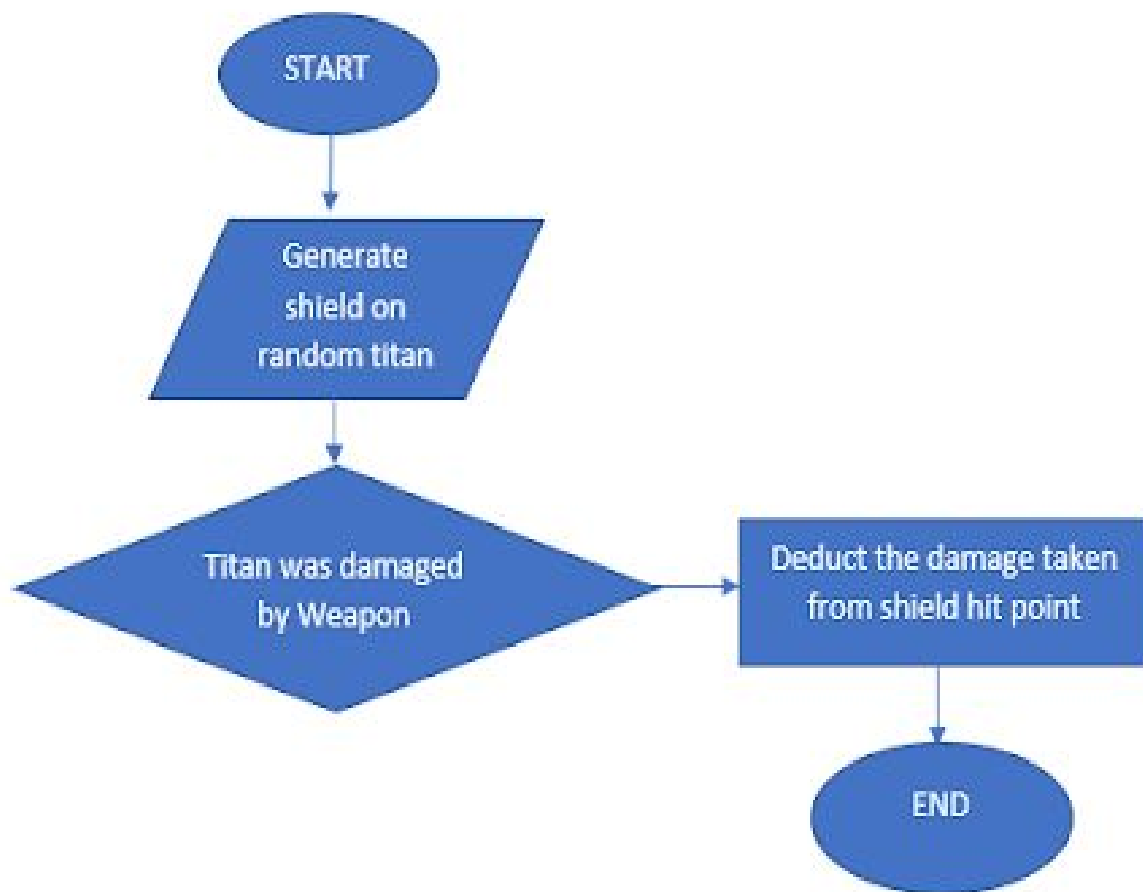
ブスの巨人をしてんりょうされる!!!!

GOING TO BE OVERRUN BY TITANS!!!

Shield.java

We created Shield class as a random shield generator as a protection to the titans along the same column as the shield in order to make the game more challenging. A shield of 10 hit points will be generated on Ground from Row 0 to Row 8. No shield is generated along Row 9. Shield hit points will be reduced by weapons first before the weapons manage to deal damage to the titans.

```
Player's turn
On The Ground
Row
0          #                                HOUR 25
1        A #                                Coin: 10
2
3
4        A
5
6
7          #          #          A
8                #
9        C          C          A
***      *** ***      *** *** ***
*** *** *** *** *** *** ***
*** *** *** *** *** *** ***
--- --- --- --- --- --- --- --- ---
 0  1  2  3  4  5  6  7  8  9      The Wall
40 20 20 20 40 40 10 20 40 50      Index
--- --- --- --- --- --- --- --- ---      HP
```

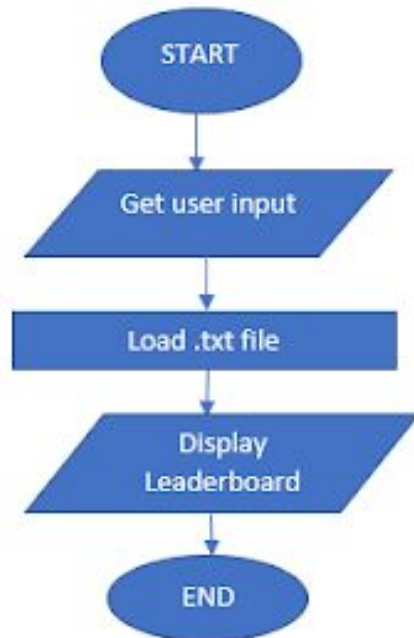


Leaderboard

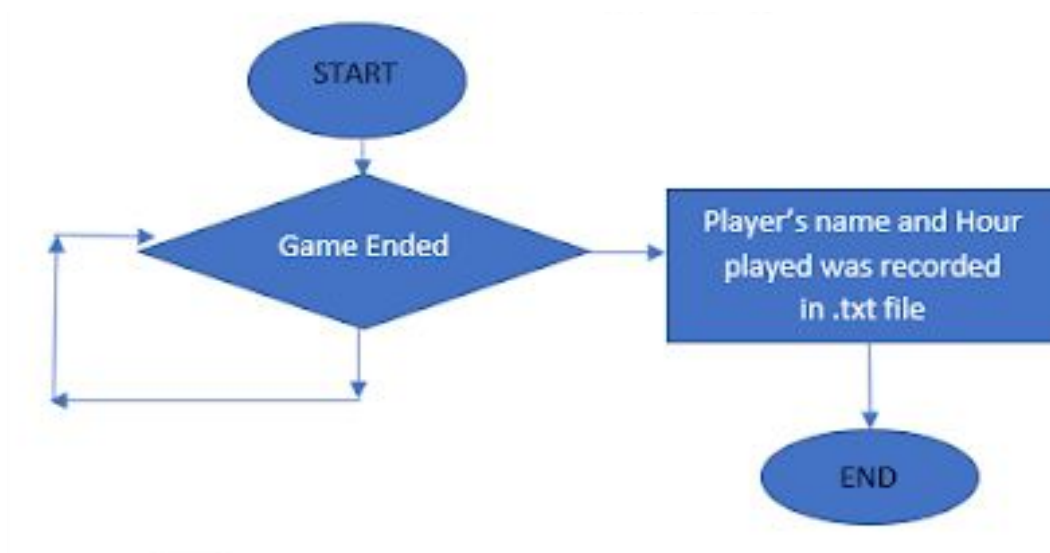
We created a Leaderboard class to keep the record of the player's gaming result. Two .txt files will be created each for endless mode and normal mode gameplay respectively. Player's name and highest hours played will be recorded. Leaderboard will always be updated before save and will always be saved before exit the game. Players can view the leaderboard via the Main Menu.

Normal Mode	
Name	Hours Survived
#1 gr	27
#2 frankie	19
#3 g	18
Endless Mode	
Name	Hours Survived
#1 fra	24
#2 cl	22
#3 gl	17
Press Enter to go back to Main Menu	

❖ Player view the Leaderboard:



❖ Leaderboard after Gameplay



Color.java

We create a Color class to change the text color in the terminal or console window using ANSI escape sequences (i.e. `TEXT_RED = "\u001B[31m"`). A static method `colorize()` is used to specify which colour we want the text to change to. For example, `Color.colorize("Hello World", "red")` will print the text in red: “Hello World”.



Sample Output

その時
At that moment

思い出した
I realised

そうだ

この世界は
that this world

Counting the coins...

OK

Writing to leaderboards...

OK

Preparing the arena...

OK

Pumping the gas...

OK

Testing the weapons...

OK

Airlock on Train

Enter player name (maximum 15 characters):

Main Menu

1. Play Game
2. Leaderboard
3. Quit

What do you want to do? (1-3):

Normal Mode

	Name	Hours Survived
#1	gr	27
#2	frankie	19
#3	g	18

Endless Mode

	Name	Hours Survived
#1	fra	24
#2	cl	22
#3	gl	17

Press Enter to go back to Main Menu

Which one do you choose? (1-3):

```
[#####]72%
```

[illegible]

```

The Game Started
Player's turn
On The Ground
Row
0
1
2
3
4
5
6
7
8
9
                                HOUR 0
                                Coin: 50

    ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
    0   1   2   3   4   5   6   7   8   9
50  50  50  50  50  50  50  50  50  50
    ---  ---  ---  ---  ---  ---  ---  ---  ---  ---

The Wall
Index
HP

Choose the weapon(s) you would like to upgrade (Type a string of integer or hit Enter to skip)
123440

```

```

Choose the weapon(s) you would like to upgrade (Type a string of integer or hit Enter to skip)
123440
Total Cost: 15
On The Ground
Row
0
1
2
3
4
5
6
7
8
9
                                HOUR 0
                                Coin: 35

                                ***
***  ***  ***  ***  ***
    ---  ---  ---  ---  ---  ---  ---  ---  ---  ---
    0   1   2   3   4   5   6   7   8   9
50  50  50  50  50  50  50  50  50  50
    ---  ---  ---  ---  ---  ---  ---  ---  ---  ---

The Wall
Index
HP

Do you want to upgrade all walls? (press 1 if yes, press Enter if no) Current coin number: 35

```

On The Ground

Row

0

HOUR 0

1

Coin: 35

2

3

4

5

6

7

8

9

*** **

---	---	---	---	---	---	---	---	---	---
0	1	2	3	4	5	6	7	8	9
50	50	50	50	50	50	50	50	50	50
---	---	---	---	---	---	---	---	---	---

The Wall

Index

HP

Do you want to upgrade all walls? (press 1 if yes, press Enter if no) Current coin number: 35

1

How many HP do you want to add up to all the walls? Current coin number: 35

2

Do you want to upgrade all walls? (press 1 if yes, press Enter if no) Current coin number: 35

1

How many HP do you want to add up to all the walls? Current coin number: 35

2

Total Cost: 20

On The Ground

Row

0

HOUR 0

1

Coin: 15

2

3

4

5

6

7

8

9

*** **

---	---	---	---	---	---	---	---	---	---
0	1	2	3	4	5	6	7	8	9
52	52	52	52	52	52	52	52	52	52
---	---	---	---	---	---	---	---	---	---

The Wall

Index

HP

Choose the wall(s) that you would like to upgrade (Type a string of integer or hit Enter to skip)

```

On The Ground
Row
0
1
2
3
4
5
6
7
8
9

***
*** *** *** *** ***
--- --- --- --- ---
0 1 2 3 4 5 6 7 8 9
52 52 52 52 52 52 52 52 52 52
--- --- --- --- ---

The Wall
Index
HP

Choose the wall(s) that you would like to upgrade (Type a string of integer or hit Enter to skip)
789

```

```

Choose the wall(s) that you would like to upgrade (Type a string of integer or hit Enter to skip)
789
How many HP do you want to add up to the wall(s)? Current coin number: 15
3
How many HP do you want to add up to the wall(s)? Current coin number: 15
3 3 3
Total Cost: 9
On The Ground
Row
0
1
2
3
4
5
6
7
8
9

***
*** *** *** *** ***
--- --- --- --- ---
0 1 2 3 4 5 6 7 8 9
52 52 52 52 52 52 55 55 55 55
--- --- --- --- ---

The Wall
Index
HP

```

```

Coin + 5
Player's turn
On The Ground
Row
0
1
2
3
4
5
6
7
8
9

***
*** *** *** *** ***
--- --- --- --- --- --- --- --- --- ---
0 1 2 3 4 5 6 7 8 9
52 52 52 52 52 52 52 55 55 55
--- --- --- --- --- --- --- --- --- ---

The Wall
Index
HP

Choose the weapon(s) you would like to upgrade (Type a string of integer or hit Enter to skip)

```

```

1 COLOSSUS TITAN APPEARED!
COLOSSUS TITAN COORDINATES :
Row 9 Column 7

Player's turn
On The Ground
Row
0
1
2
3
4
5
6
7
8
9

A
C

***
*** *** *** *** *** *** ***
--- --- --- --- --- --- --- --- --- ---
0 1 2 3 4 5 6 7 8 9
52 52 52 52 52 52 52 55 55 55
--- --- --- --- --- --- --- --- --- ---

The Wall
Index
HP

Choose the weapon(s) you would like to upgrade (Type a string of integer or hit Enter to skip)

```

On The Ground

Row

0

HOUR 15

1

A

Coin: 125

2

3

4

5

A

6

7

8

9

CC

A

---	---	---	---		---	---	---	---	---	
0	1	2	3	4	5	6	7	8	9	
50	50	50	40	0	10	30	40	50	50	
---	---	---	---		---	---	---	---	---	

The Wall

Index

HP

!!!!!!!

か。か。。。壁に穴挙げられた

They blew a h-hole into the w-w-wall.....

いってくるぞ!

They're coming!

巨人たちが入ってきた!!!!

THE TITANS ARE COMING!!!

ダメなんだ。。。。

It's all over.

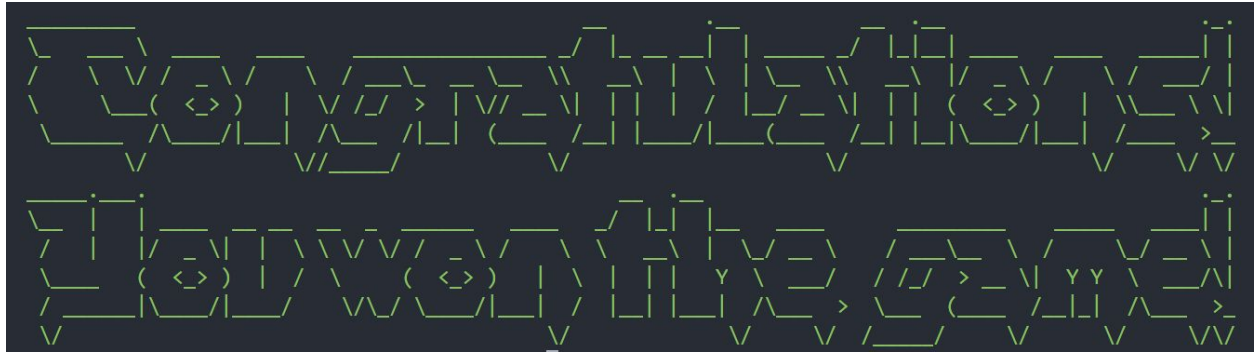
この町を

This town is...

もう

ブスの巨人をしてんりょうされる!!!

GOING TO BE OVERRUN BY TITANS!!!



Hour(s) Survived : 25
Number of Armoured Titan(s) killed: 20
Number of Colossus Titan(s) killed: 15
Number of shield(s) destroyed : 0

Press Enter to exit!