**DOKUZ EYLUL UNIVERSITY**

**ENGINEERING FACULTY**

**DEPARTMENT OF COMPUTER ENGINEERING**

**CME1252 PROJECT BASED LEARNING – II**

**FINAL REPORT**

**PROJECT – II**

**Gravity**

**by**

**Arda Aydın**

**Emil İsmayilzada**

**Kerem Kalıntaş**

**Burak Kıyak**

**Lecturers**

**Göksu Tüysüzoğlu**

**Derya Birant**

**IZMIR**

**28.04.2023**

# CHAPTER ONE

PROGRESS DESCRIPTION

In this project, we had to develop a console app, more accurately a game. In order to develop this game, we worked as a team, and I would say we worked perfectly as a team. We often had meetings together, and we talked through the algorithms we used and problems we encountered. In the first week we discussed upon the main design of the game, then designed some of the needed classes, and along the way we finished initializing the game and filling the map. Passing to second week, we successfully done the player movements, timing and backpack. In the third week, we mainly worked on the Input Queue. The fourth week was one of the most essential weeks, since we finished doing robot and boulder movements. And of course, we made the debugging and perfected the appearence of our project.

# CHAPTER TWO

TASK SUMMARY

## Completed Tasks

Arda Aydın : Even though I have a lot of contributions to so many different parts of this project, I can still give some specific examples. Especially I worked in player movements, robot movements, input queue and bacpack stack.

Emil Ismayilzada: I would easily say that I was part of the team in this project. I worked in many parts of the project, and helped my friends. Especially I worked in

making and filling inside of the map. Aside from that I worked on the algorithm of the boulders. Also I helped in robot movements, player movements, input queue and etc. And of course I edited the video and prepeared the presentation.

Burak Kıyak: To give some examples, I worked in writing necessary functions,the endgame menu, startgame menu, player movements and boulder movement, boulder push and of course in the debugging proccess. However, these are just few examples. I participated very decisively in this project and helped all of my group mates, when needed.

Kerem Kalıntaş: I played a major role in the making of the project. I took part in most of the parts. But as an instance, I would say I contributed a lot in the timing and the pace of the game. I also wrote inputQueue class and “replaceRandomSquare”, “printStars”, “initGame” functions. Apart from these, I always tried to be a team player and assisted my teammates.

## Incomplete Tasks: Reasons and Explanations

I do not think we have any incompleted tasks in our project. Instead we had lots of additional features

## Additional Improvements ~~to the Project~~

Arda Aydın : We added a beatiful “Game Over” text at the end of the game.

Emil Ismayilzada : We made lots of additional improvements to UI of our project.

Kerem Kalıntaş: We printed stars for the background of the game. And added a feature to ask user if he/she wanted to play again.

Burak Kıyak: We added a “Gravity” text and stars in our menu.

# CHAPTER THREE

EXPLANATION of algorıthms

## Screenshots

## Functions

These are the function we created:

void mainMenu();

void initGame(String mapFilePath);

boolean runGame();

void fallBoulder(Boulder boulder);

Robot moveRobot(Robot robot);

void teleportPlayer();

void movePlayer(int moveX, int moveY);

void addToBackpack(char newNumber);

void setCursorPos(int x, int y);

void setConsoleColor(Color foregound, Color background);

void setSquareColor(char square);

void setSquare(int squareX, int squareY, char square);

void printPlayerStatus();

void setRandomEarth(char square);

void replaceRandomSquare(char[] searchSquares, char newSquare);

void printStars(int width, int height);

## Algorithms and Solution Strategies

Arda Aydın :

First, we check whether the item the user is trying to add to the backpack is empty, the same as the one in the backpack, or already full, using if-else structures. If it is the same, we proceed with the necessary scoring. If it is already full, we delete the top item in the backpack and add the new item. If the number under the added and deleted items is the same, the scoring is adjusted accordingly. If it is empty, we simply add the item. Then, we perform a cleaning process by writing blank spaces to the designated display location. Next, we use a temporary stack to perform addition and subtraction operations and print them to the screen. Finally, we display the score on the screen.

Emil Ismayilzada: We came up with lots of algorithms and solution strategies in this project as a team. I say as a team, because even though one of our team member came up with an idea, we improved and worked on this idea as a real team. For instance, in this project, we were required to add a new element from the input queue in every three seconds. Firstly, we created a seperate class for input queue, which contains functions like: print (prints the elements in the queue), next(adds a new element to the queue), generateSquare(generates random elements for their probibility out of 40). After that, in gravity class, we said that if inputQueueTimer is bigger than 3000, then add element. And of course, for in this adding proccess we did the required convertion operations with switch cases.

Burak Kıyak: In ending the game we don’t know anything about enigma so I don’t know how to clear enigma. Then I convert map to empty square then I print.

Another algorithm is falling boulder. When I do that I had to pay attention to the order. Firstly I checked under the boulder is empty square or not. Secondly I check under is boulder or not. Lastly I checked player.Then I did movements of boulder.

Kerem Kalıntaş: I wrote the “replaceRandomSquare” method to replace random squares with the given parameters. I created this method for replacing squares when the input queue is changed. It also functions as replacing random earth squares with a given square type. So it is being used in the “initGame” method also. I also wrote the “printStars” method to render stars as the background of the game. This function iterates the whole screen and prints a random star with a random color with the 5% posiblity. And prints a is a space char with the 95% posiblity.

# CHAPTER FOUR

PROBLEMS ENCOUNTERED

Arda Aydın : Bilgisayar hareketlerini yaparken bazen bilgisayar istenilen gibi hareket etmiyordu ve “X” hareket edecek yeri kalmayınca oyun donuyordu.

Emil Ismayilzada: In boulder movements, first we tried an algorith, where we scaned all of the map from above, and if there were any boulders, they’er moved one digit down. However, because we were scanning from above, the program was moving the boulders again and again. This was one the many annoying problems we encountered in this project.

Kerem Kalıntaş: Enigma window didnt shutdown after returning from the main. We wanted it to close when the user didnt wanted to play anymore. As the solution we wrote “System.exit(0)” instead of returning from main.

Burak Kıyak: Firstly I want to boulder movements in boulder class. But I can’t understand why am I getting outofbounds error. Then I decided to do it searching boulders in map with nested fors.

# CHAPTER FIVE

conclusıon

In this project, we had drastic improvement at our knowledge in Java and OOP. We perfected our abilities of using stacks, queues, classes and etc. However, we did not just enhance our technical knowledge. Indeed, we learned, overcoming the hardships as a team, deciding on crucial decisions as a team, generally working as a team. Other than that, we gained an understanding on preparing reports, presentations and posters, which will help us in our professional careers.

REFERENCES

HTTPS://LEARN.MICROSOFT.COM/ENUS/DOCS/ HTTPS://EN.WIKIPEDIA.ORG/WIKI/WIKIPEDI A HTTPS://STACKOVERFLOW.COM/ HTTPS://WWW.AMAZON.COM/INTRODUCTIO N-OBJECT-ORIENTED-PROGRAMMINGJAVA/DP/0073523305 R