INTEGRATED PROJECT REPORT

On

TETRIS GAME

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CSP2208) of

COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2016

in

May-2019

Under the Guidance of	
Name of the Project Guide	**Jasmeet Kaur**
Designation of the Project Guide	**Assistant Professor**
Submitted By	**Gaurav Walia** **Roll. No. 1610991300** **Gunbir Plaha** **Roll. No. 1610991309** **Gagan** **Roll. No. 1610991292**

SCHOOL OF COMPUTER SCIENCES
CHITKARA UNIVERSITY
PUNJAB

INTEGRATED PROJECT REPORT

On

TETRIS GAME

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CSP2208) of

COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2016

in

May-2019

Under the Guidance of	
Name of the Project Guide	**Jasmeet Mam**
Designation of the Project Guide	**Assistant Professor**
Submitted By	**Gaurav Walia** **Roll. No. 1610991300** **Gunbir Plaha** **Roll. No. 1610991309** **Gagan** **Roll. No. 1610991292**

SCHOOL OF COMPUTER SCIENCES
CHITKARA UNIVERSITY
PUNJAB

CERTIFICATE

This is to be certified that the project entitled TETRIS Game has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2019 - May-2019 is a bonafide piece of project work carried out by Gaurav Walia (1610991300), Gunbir Plaha (1610991309) and Gagan (1610991292) of the students group towards the partial fulfillment for the award of the course Integrated Project CSP228 under the guidance of Jasmeet Mam and her supervision.

(Signature)
Jasmeet Kaur
(Assistant Professor)
(Computer Science Department)

CANDIDATE'S DECLARATION

We, **Gaurav Walia (1610991300), Gunbir Plaha (1610991309) and Gagan (1610991292) OF THE STUDENTS GROUP,** B.E.-2016 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled **TETRIS GAME** is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

(Signature)	(Signature)	(Signature)
Gaurav Walia	Gunbir Plaha	Gagan
1610991300	1610991309	1610991292

Place:
Date:

ABSTRACT

The issue for the Gamer which use Linux Distribution have to find the games for their Linux Distribution. Therefore we created a prototype of a Tetris Game that will save time for the Linux Users to build and play the game easily. The Tetris Game is a game in which different shapes come down one after another and we have to collect these shapes by performing operations (i.e moving down, moving right, moving left and rotating shape) at the bottom so that no space left. When a row get fully filled then the row is erased and we get a point. Although this Tetris Game has old features but it has some new features for the Gamer like Individual Account Creation, Game Saving Options, Changing User Interface of the game, etc. Moreover a user can compare his performance with another users also. This game is open sourced i.e available for others developers to improve and add new features to it on Git Hub. The Game has written in JAVA programming language and we have followed the Object Oriented Features like Abstraction, Encapsulation, etc. The project management have done using scripting languages like BASH and Power Shell. We have also used Version Control System i.e git for maintaining our software easily. So our game is very useful for beginners who use Linux and want to play Games.

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to ***Jasmeet Kaur*** for his support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

We also extend our sincere appreciation to ***Jasmeet Kaur*** who provided his valuable suggestions and precious time in accomplishing our Integrated project report.

Lastly, We would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to day experience and received lots of suggestions that improve our quality of work.

CONTENTS

LIST OF IMAGES

INTRODUCTION

Statement of the Problem

Our aim is to develop a Java Based CLI Game which will run on Cross Platform Devices.This game has many options for the gamer such as *Individual Game Saving Feature* which most games don't have. In this game we have *User Comparison Feature* which will compare two user with their performance. Also the users have *UI Color Change Control* to make their own game interface according to their choice.

Background

Nowadays the Linux Gamers are facing problems so we have solved that problem by making a TETRIS Game which is easy to build and run on many devices.

Purpose of Project and Overview of Project Report

Our purpose of project is creating a CLI based game for Linux/Unix Users and Windows Users. In this project we have created a TETRIS Game in which firstly user is prompted some options of the Game like Exit, Start new game, Start Existing Game, etc. Then when user starts a new game then user have to type the size of the game i.e rows and cols. After that the specified size game board is created and user is ready for playing. There are 4-5 controls that user can perform i.e down, right, left, save and exit. In this game user can change the User Interface according to his/her choice which makes it interactive.

Image I

SOFTWARE REQUIREMENT SPECIFICATION

Methods and Design Approach

We have designed full application using Object Oriented Principles like Abstraction, Encapsulation, Inheritance, etc. We have also used Exception Handling and File Handling for game interaction and user experience.

Programming/Working Environment

We have used JAVA Programming Language for writing this game and we have also used scripting languages like Bash and Power Shell. For working environment we have used Visual Studio Code.

Requirements to run the application

For running this game application Linux or Unix is recommended, but it can also run on Windows CMD and *javac* compiler is needed for newer builds. *RAINBOW package* is also required for better UI and UX in Linux / Unix.

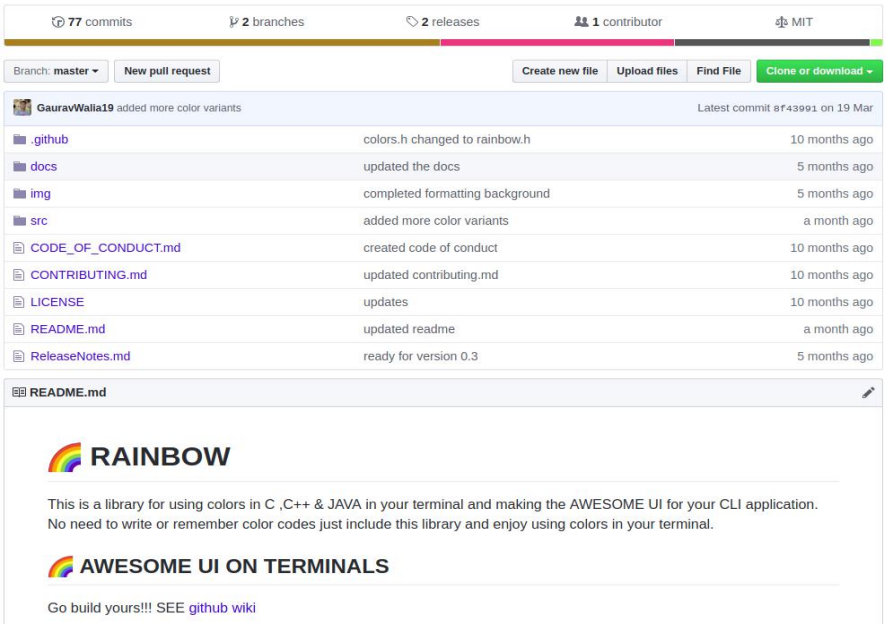

Image II

PROJECT PLAN

The major aspect of the work has been to develop a proper logic and Object Oriented Design of the Tetris Game. File Handling is done as to make a better utilization of the resources available and also used for good performance.

A detailed study of Java and its Object Oriented Principles was conducted as an initial step.

The study of Java Security Package classes have also helped in our password checking development work.

We have used Test Driven Development in developing this game. But we have developed the project according to these 4 stages:

- Information Gathering
- Design
- Implementation
- Testing

1. INFORMATION GATHERING

Firstly we have collected some information about the game such as its game play, game working, etc. Then we have collected and think ideas about this game i.e how we can make it good for the users. After the ideas collection we have also collected information about the better user interface in Command Line Interface so we have used RAINBOW package for better User Interface.

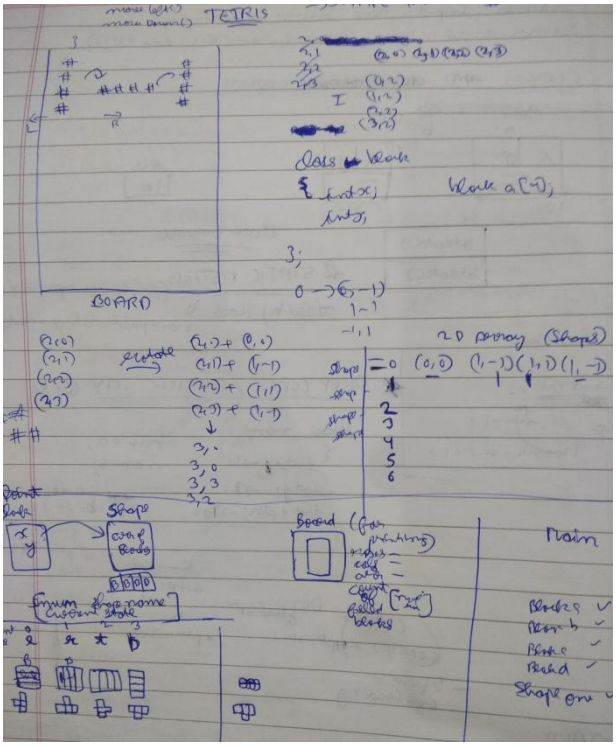

Image III

2. DESIGN

Second step we followed i.e to design the whole game which was very though and important task for an application but at the end we developed the whole design by self. Firstly we have designed this project class by class and at the end we designed the whole application for game and work flow. Designing file structure was also very important task which we have designed using Test Driven Development i.e designed some test cases before development.

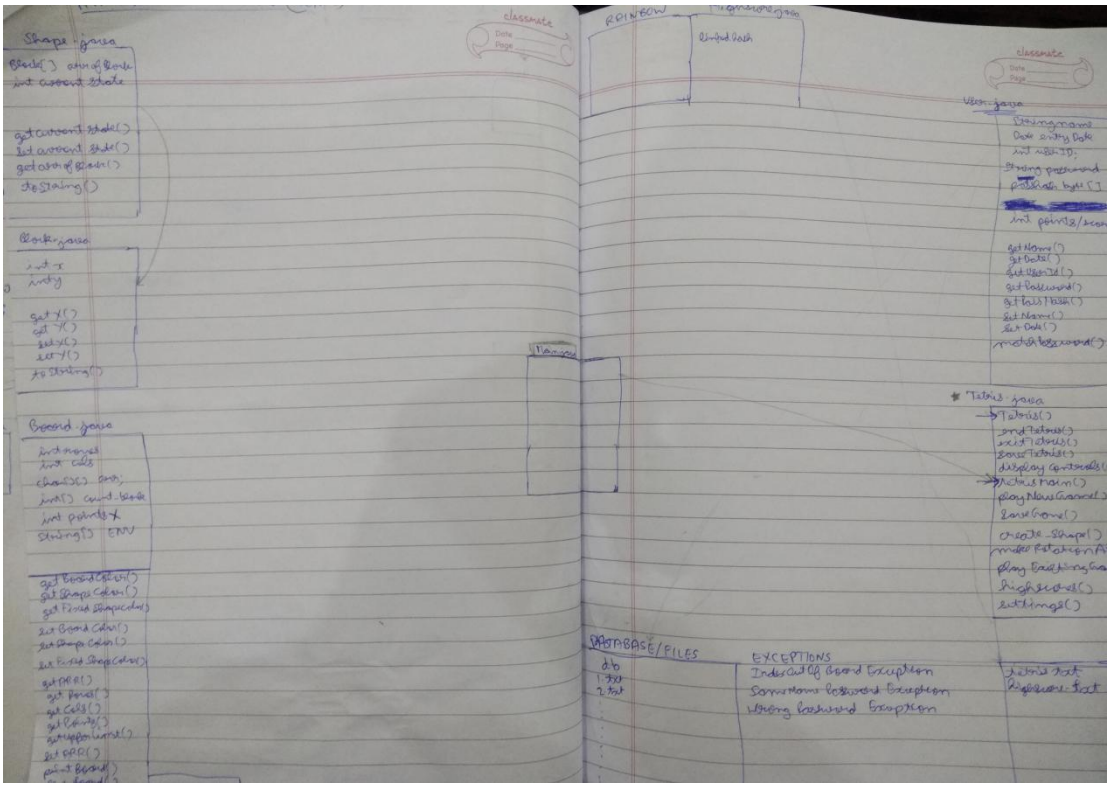

Image IV

3. IMPLEMENTATION

The third step was to implement the whole application in JAVA which was a very big task for us but we have implemented it step by step by developing new class by class. While implementing we have updated design for better efficiency. Moreover for software maintenance and version control we have used Version Control System like git and hosted an open repository on Git Hub. For combining some tasks we have also used scripting languages like BASH for Linux and Power Shell for windows.

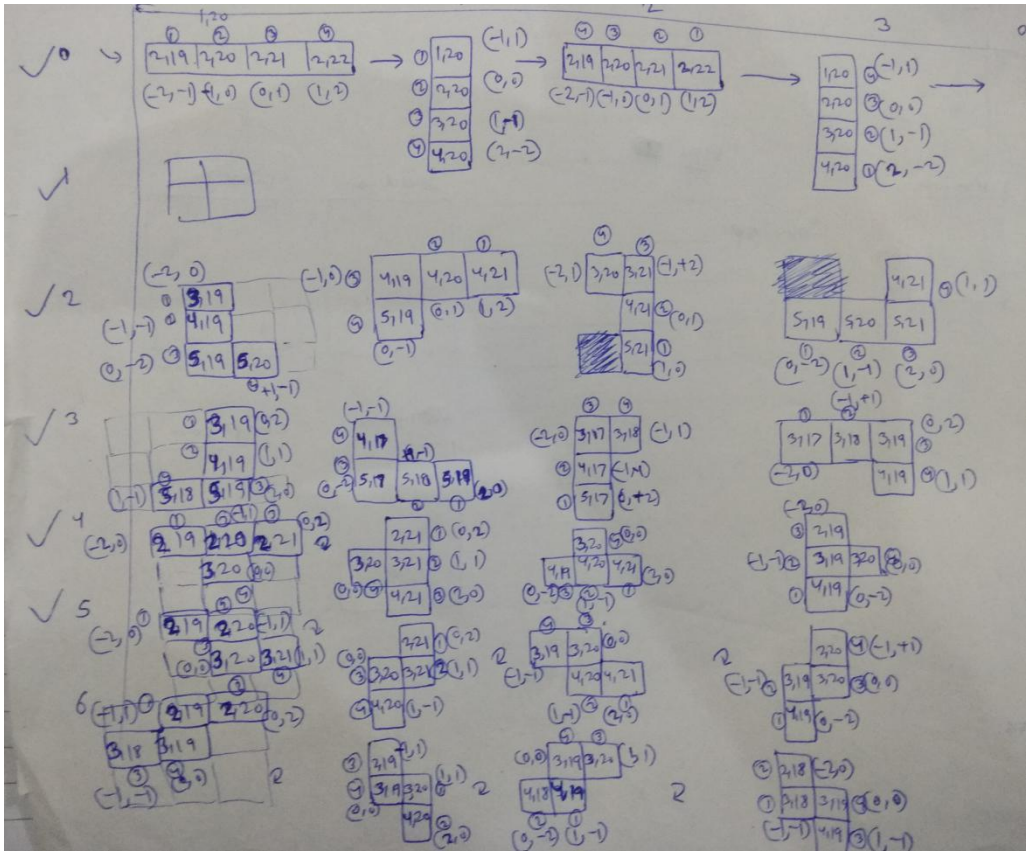

Image V

4. TESTING

For testing our application we have manually tested it. Also we have built a script for locally testing the game. For Continuous Integration (CI) we have used travis that can test the build of application automatically.

Image VI

In this way we planned this project step by step so that we cannot face issues while development.

FILE HANDLING DESIGN AND IMPLEMENTATION

In this Game Application instead using Data Bases we have used File Handling for storing data of the user. Also for storing some variables like id counter for future use we have used file for storing that information.

DESIGN

This Tetris Game is designed using files like *tetris.txt* for storing id counter, *highscore.txt* for storing highscores, *history.txt* for storing user history, *default.txt* for storing defaults and directory db which has user files with names with ids like *1.txt*, *2.txt*, etc.

Image VII

IMPLEMENTATION

This Game Application is implemented using *BufferedReader*, *FileReader*, *BufferedWriter* and *FileWriter* classes in JAVA.

```
File file = new File(filename);
file.getParentFile().mkdirs();
file.createNewFile();

BufferedWriter geeks_out1 = new BufferedWriter(new FileWriter(file));

/**
 * saving user info
 **/
geeks_out1.write(U.getName()+"|"+U.getDate()+"|"+U.getUserId()+"|"+U.getPassHas

for(int i=0;i<4;i++)
{
    int x = LINE.getarrofblock()[i].getX();
    int y = LINE.getarrofblock()[i].getY();
    geeks_out1.write(x +"|"+y+"|");
}
geeks_out1.write("\n");                                                    //
geeks_out1.write(LINE.getcurrentstate()+"|"+shape_counter+"|\n");

//saving env variables
String[] envcolors = B.getENV();
```

Image VIII

Program's Structure Analyzing and GUI Constructing

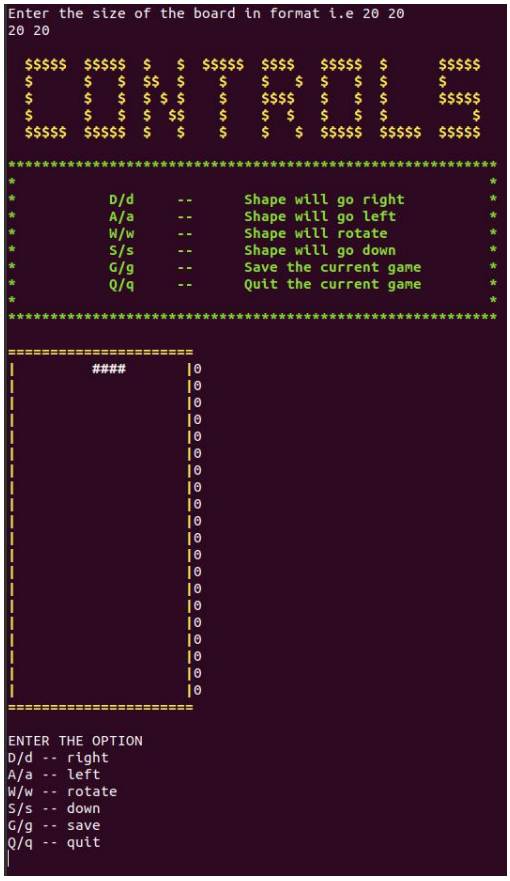

Image IX

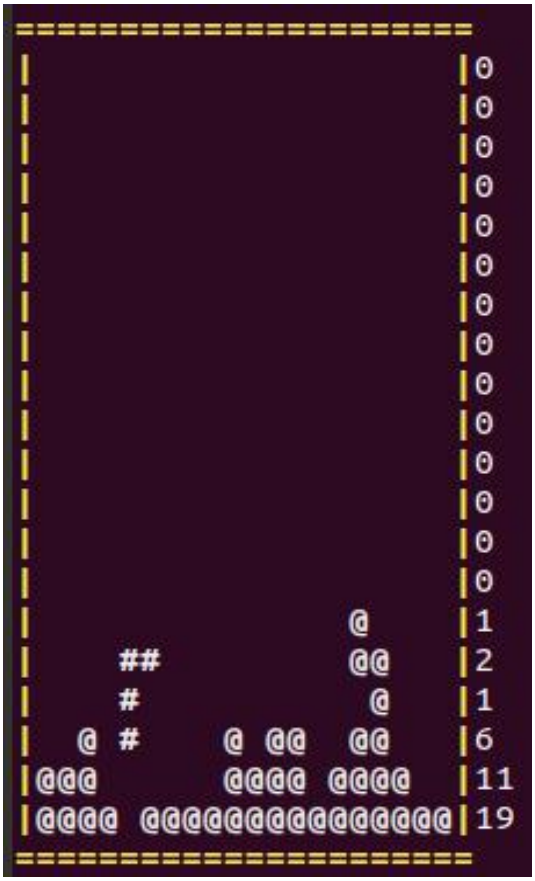

Image X

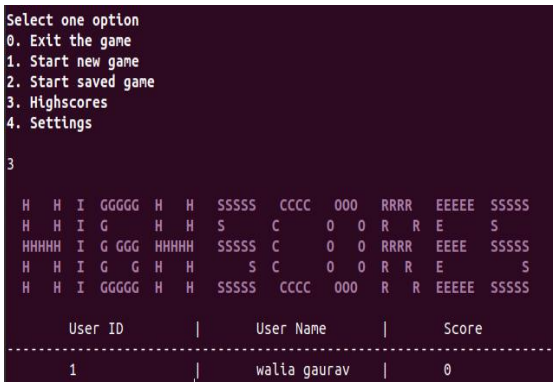

Image XI

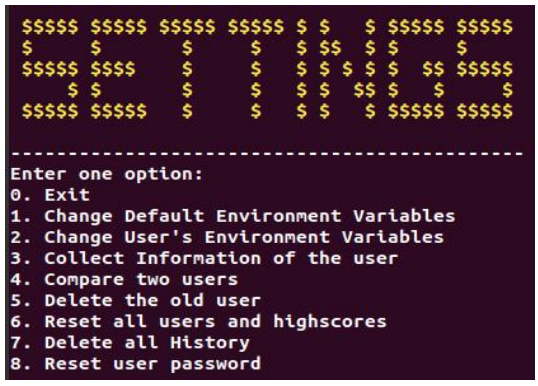

Image XII

Code-Implementation

```java
/**
 * This function is displays the main functions in tetris game
 *
 * @return void
 **/
public void tetrisMain()
{
    Scanner in = new Scanner(System.in);                                    // scanner for input in tetr
    Rain color = new Rain();                                                // for using rainbow
    System.out.println(color.BOLD);
    System.out.println("Select one option");
    System.out.println("0. Exit the game");
    System.out.println("1. Start new game");
    System.out.println("2. Start saved game");
    System.out.println("3. Highscores");
    System.out.println("4. Settings");
    System.out.println(color.RESET);
    try
    {
        int option = in.nextInt();
        switch(option)
        {
            case 0:
                exitTetris(color.BDGREEN);
                System.exit(0);
            case 1:
                playNewGame();
                break;
            case 2:
                playExistingGame();
                break;
            case 3:
                scorelist.displayHighScore();
                break;
            case 4:
                settings();
                break;
            default:
                throw new InvalidOptionException("Invalid Option entered by the user");
        }
```

Image XIII

```java
import java.io.*;
import java.util.*;
You, 19 hours ago | 1 author (You)
/**
 * Class for making the user
 **/
public class User
{
    private String name;            // name of the user
    private Date entryDate;         // storing entry date of the user
    private int userId;             // for user ids
    private String password;        // password of the user
    private String passhash;        // sha256 hash for password
    private int score;              // score made by the user

    /**
     * This constructor will create the new user
     *
     * @param String name of the user
     * @param String password of the user
     * @param Date of entrydate of the user
     * @param idsetter for setting and maintaining the id
     **/
    public User(String name,String password,Date entryDate,String passhash,int score)
    {
        this.name = name;
        this.entryDate = entryDate;
        this.password = password;
        this.passhash = passhash;
        this.score = score;
        File file = new File("tetris.txt");                                 // create a file ob

        try
        {
```

Image XIV

```java
import RAINBOW.*;
import java.io.*;
You, 4 days ago | 2 authors (You and others)
/**
 * Board class for making board
 **/
public class Board
{
    private int rows;                                   // rows of the board
    private int cols;                                   // cols of the board
    private char[][] arr;                               // array for making board
    private int[] count_block;                          // store count the number o
    private int points;                                 // points of collecting dur
    private String[] ENV;                               // array for environment va

    /**
     * Parameterised Constructor for making new board
     *
     * @param int for rows
     * @param int for cols
     **/
    public Board(int rows,int cols)
    {
        this.rows=rows;
        this.cols=cols;
        arr = new char[rows][cols];                     // create dynamic array for
        count_block = new int[rows];                    // set array for storing co
        points=0;                                       // initially scored points
        for(int i=0;i<rows;i++)
        {
            for(int j=0;j<cols;j++)
            {
                arr[i][j]=' ';                          // setting space in the boa
            }
```

Image XV

```java
public class Shape
{
    private Block[] arrofblock;
    private int currentstate;

    public Shape(Block a,Block b,Block c,Block d,int currentstate)
    {
        arrofblock = new Block[4];
        this.arrofblock[0] = a;
        this.arrofblock[1] = b;
        this.arrofblock[2] = c;
        this.arrofblock[3] = d;
        this.currentstate = currentstate;
    }
    //getters
    public int getcurrentstate()
    {
        return currentstate;
    }       Gaurav Walia, 7 months ago • MOVING SHAPE DOWN

    /**
     * This will set new state
     *
     * @param state for setting state
     * @param void
     **/
    public void setcurrentstate(int state)
    {
        this.currentstate = state;
    }

    /**
     * This function will return the block array of a shape
     *
     * @return blocks array
     **/
    public Block[] getarrofblock()
    {
        return arrofblock;
    }
```

Image XVI

LIMITATIONS

Although our project is good but it has limitations like it is only Command Line Interface Game. Also it has old types of key pressing controls as compared to new games. The game has larger size for CLI i.e 50-60 kb. User Interaction is not good on Windows as compared to Linux / Unix.

CONCLUSIONS

Therefore we concluded that the work of developing this CLI Based Tetris Game is a project work which keeps on taking its own course along the whole timeline. We are having fixed in our minds certain targets which we have enumerated before in this report. So, we find it important to take stock of the whole work that we have done till now. The understanding of the Object Oriented Principles in JAVA has really helped us lot in making this project.

FUTURE SCOPE

The future scope of this project is that it will not remained CLI based Application but it will run using GUI frameworks of JAVA like Swing, Applet, etc. Also it will be used with JGAME Project which is Open Source.

REFERENCES

- google.com
- wikipedia.com
- github.com
- Quora.com
- Slideshare.com