Design and Development of Rock-Paper-Scissors Game using Java

Team No.10

6830300088 เจตนิพัทธ์ พรหมศร

6830300631 ภัทราภรณ์ นาคสิทธิ์

6830300908 เสฏฐพงศ์ โพธิ์ศรี

Submit to

Asst.Prof.Dr. Kullawadee Somboonviwat

03603112 Programming Fundamentals

สารบัญ

Rock-Paper-Scissors Program

Rules of the rock-paper-scissors game

เกมเป่ายิ้งฉุบเป็นเกมที่ผู้เล่นเลือกหนึ่งในสามตัวเลือก ได้แก่ ค้อน(Rock),กระดาษ (Paper),กรรไกร (Scissors) มีกติกาดังนี้

— ค้อน ชนะ กรรไกร

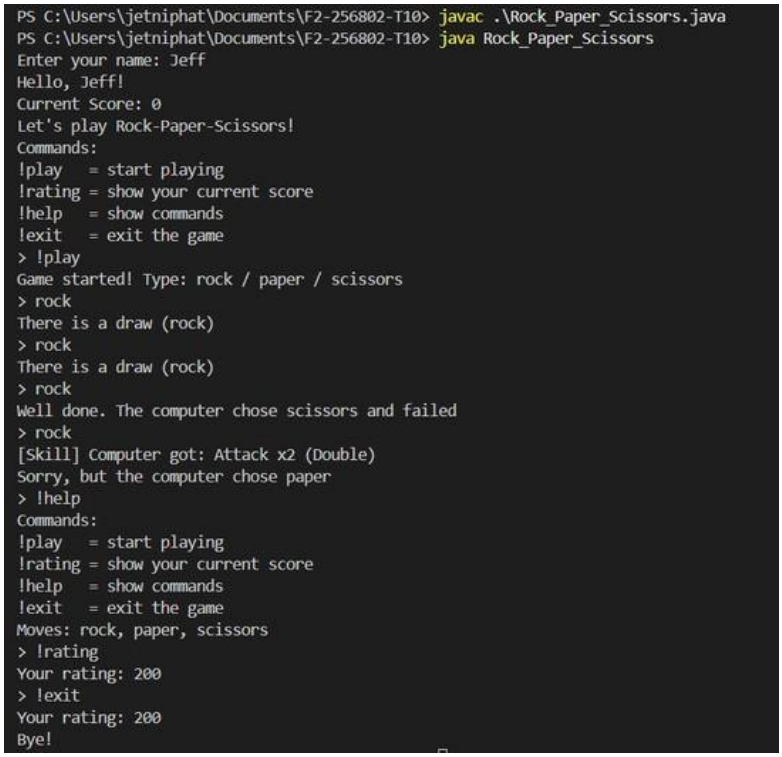
— กรรไกร ชนะ กระดาษ

— กระดาษ ชนะ ค้อน

— หากผู้เล่นและคอมพิวเตอร์เลือกเหมือนกัน จะถือว่าเสมอ

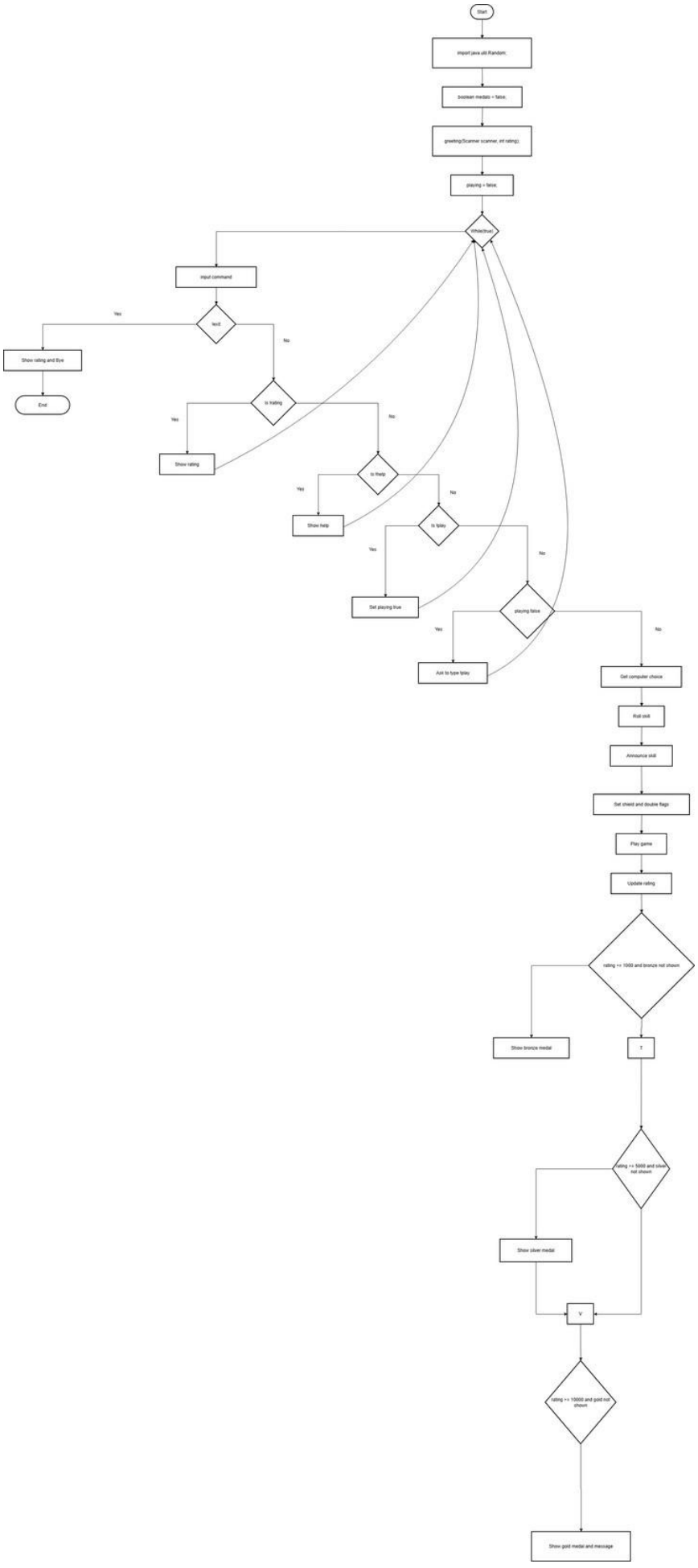
Sample runs

ตัวอย่างผลลัพธ์ที่ได้จากการรันโปรแกรมแสดงให้เห็นการเลือกของผู้เล่นและคอมพิวเตอร์ รวมถึงผลลัพธ์ของเกม



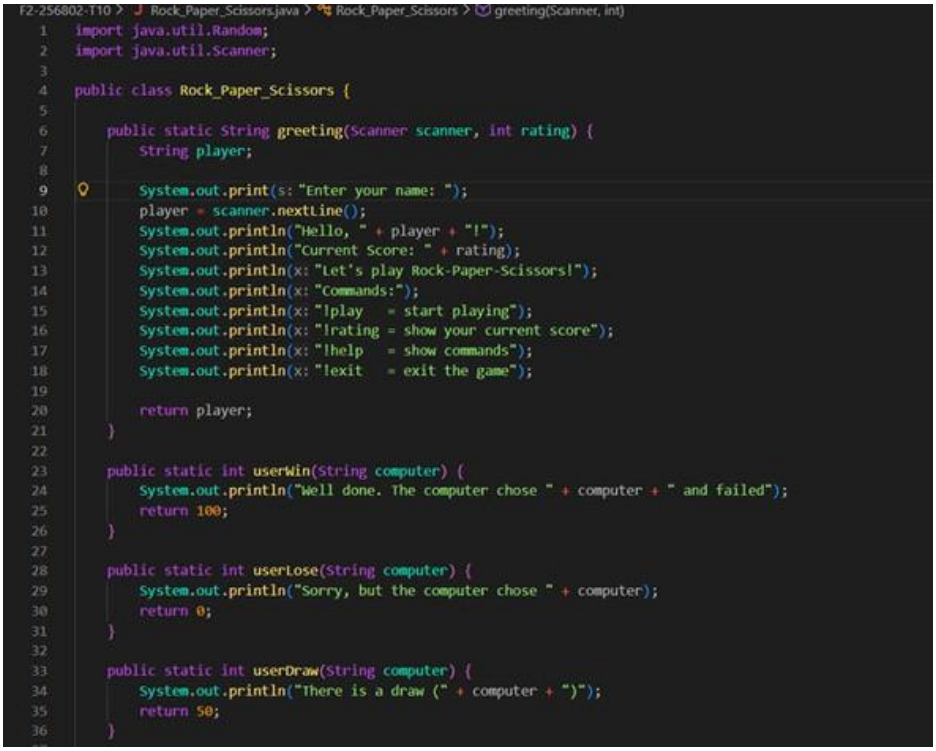
Program Design

Flowchart



Program Design

Program Structure



```java

    public static int[] rollSkill(Random random) {
        int skillOwner = 0;
        int skillType = 0;

        int chance = random.nextInt(bound: 100);
        if (chance < 30) {
            skillOwner = random.nextBoolean() ? 1 : 2;
            skillType = random.nextBoolean() ? 1 : 2;
        }

        return new int[]{skillOwner, skillType};
    }

    public static void announceSkill(String playerName, int[] skill) {
        int owner = skill[0];
        int type = skill[1];

        if (owner == 0)
            return;

        String who = (owner == 1) ? playerName : "Computer";
        if (type == 1) {
            System.out.println("[Skill] " + who + " got: AI attack has no effect (Shield)");
        } else if (type == 2) {
            System.out.println("[Skill] " + who + " got: Attack x2 (Double)");
        }
    }

    public static int playGame(String playerChoice, String computer,
                               boolean userShield, boolean aiShield, boolean userDouble) {

        int rating = 0;
```

```java
public class Rock_Paper_Scissors {
    public static int playGame(String playerChoice, String computer,

        if (playerChoice.equals(computer)) {
            return userDraw(computer);
        }

        boolean userWins = false;
        boolean userLoses = false;

        if (playerChoice.equals(anObject: "rock")) {
            if (computer.equals(anObject: "scissors")) userWins = true;
            else if (computer.equals(anObject: "paper")) userLoses = true;
        } else if (playerChoice.equals(anObject: "paper")) {
            if (computer.equals(anObject: "rock")) userWins = true;
            else if (computer.equals(anObject: "scissors")) userLoses = true;
        } else if (playerChoice.equals(anObject: "scissors")) {
            if (computer.equals(anObject: "paper")) userWins = true;
            else if (computer.equals(anObject: "rock")) userLoses = true;
        } else {
            System.out.println(x: "Invalid input!");
            return 0;
        }

        if (userWins) {
            rating = userWin(computer);
            if (userDouble) {
                rating *= 2;
                System.out.println(x: "[Skill Effect] Your attack is x2! Bonus doubled.");
            }
        } else if (userLoses) {
            if (userShield) {
                rating = userDraw(computer);
                System.out.println(x: "[Skill Effect] Shield activated! AI attack has no effect.");
            } else if (aiShield) {
                rating = userDraw(computer);
```

```
4   public class Rock_Paper_Scissors {
66      public static int playGame(String playerChoice, String computer,
104                 System.out.println(x: "[Skill Effect] Computer's Shield activated! Your attack has no effect.");
105             } else {
106                 rating = userLose(computer);
107             }
108         }
109
110         return rating;
111     }
112
113     public static String getComputerChoice(Random random) {
114         String[] computerChoices = {"rock", "paper", "scissors"};
115         return computerChoices[random.nextInt(computerChoices.length)];
116     }
117
118     public static void printHelp() {
119         System.out.println(x: "Commands:");
120         System.out.println(x: "!play   = start playing");
121         System.out.println(x: "!rating = show your current score");
122         System.out.println(x: "!help   = show commands");
123         System.out.println(x: "!exit   = exit the game");
124         System.out.println(x: "Moves: rock, paper, scissors");
125     }
126
        Run | Debug
127     public static void main(String[] args) {
128
129         int rating = 0;
130         Scanner scanner = new Scanner(System.in);
131         Random random = new Random();
132
133         boolean bronzeShown = false;
134         boolean silverShown = false;
135         boolean goldShown = false;
136
```

```
4   public class Rock_Paper_Scissors {
127     public static void main(String[] args) {
137         String playerName = greeting(scanner, rating);
138         boolean playing = false;
139
140         while (true) {
141             System.out.print(s: "> ");
142             String choice = scanner.nextLine().toLowerCase();
143
144             if (choice.equals(anObject: "!exit")) {
145                 System.out.println("Your rating: " + rating);
146                 System.out.println(x: "Bye!");
147                 break;
148             }
149
150             if (choice.equals(anObject: "!rating")) {
151                 System.out.println("Your rating: " + rating);
152                 continue;
153             }
154
155             if (choice.equals(anObject: "!help")) {
156                 printHelp();
157                 continue;
158             }
159
160             if (choice.equals(anObject: "!play")) {
161                 playing = true;
162                 System.out.println(x: "Game started! Type: rock / paper / scissors");
163                 continue;
164             }
165
166             if (!playing) {
167                 System.out.println(x: "Type !play to start playing (or !help for commands).");
168                 continue;
169             }
170
```

```
4   public class Rock_Paper_Scissors {
127     public static void main(String[] args) {
170
171             String computer = getComputerChoice(random);
172
173             int[] skill = rollSkill(random);
174             announceSkill(playerName, skill);
175
176             boolean userShield = (skill[0] == 1 && skill[1] == 1);
177             boolean aiShield = (skill[0] == 2 && skill[1] == 1);
178             boolean userDouble = (skill[0] == 1 && skill[1] == 2);
179
180             rating += playGame(choice, computer, userShield, aiShield, userDouble);
181
182             if (rating >= 1000 && !bronzeShown) {
183                 System.out.println(x: "+---------------------------+");
184                 System.out.println(x: "|         BRONZE MEDAL!     |");
185                 System.out.println(x: "|   You reached 1000 points  |");
186                 System.out.println(x: "+---------------------------+");
187                 bronzeShown = true;
188             }
189
190             if (rating >= 5000 && !silverShown) {
191                 System.out.println(x: "+---------------------------+");
192                 System.out.println(x: "|         SILVER MEDAL!     |");
193                 System.out.println(x: "|   You reached 5000 points  |");
194                 System.out.println(x: "+---------------------------+");
195                 silverShown = true;
196             }
197
```

```
4   public class Rock_Paper_Scissors {
127     public static void main(String[] args) {
178             boolean userDouble = (skill[0] == 1 && skill[1] == 2);
179
180             rating += playGame(choice, computer, userShield, aiShield, userDouble);
181
182             if (rating >= 1000 && !bronzeShown) {
183                 System.out.println(x: "+---------------------------+");
184                 System.out.println(x: "|         BRONZE MEDAL!     |");
185                 System.out.println(x: "|   You reached 1000 points  |");
186                 System.out.println(x: "+---------------------------+");
187                 bronzeShown = true;
188             }
189
190             if (rating >= 5000 && !silverShown) {
191                 System.out.println(x: "+---------------------------+");
192                 System.out.println(x: "|         SILVER MEDAL!     |");
193                 System.out.println(x: "|   You reached 5000 points  |");
194                 System.out.println(x: "+---------------------------+");
195                 silverShown = true;
196             }
197
198             if (rating >= 10000 && !goldShown) {
199                 System.out.println(x: "+-------------------------------------------------------+");
200                 System.out.println(x: "|                    GOLD MEDAL!                        |");
201                 System.out.println(x: "|              You reached 10000 points                 |");
202                 System.out.println(x: "|                                                       |");
203                 System.out.println(x: "|  Ai: You're really boring, you know.                  |");
204                 System.out.println(x: "|  But it's not like I'm playing with you because       |");
205                 System.out.println(x: "|  I like you or anything!                              |");
206                 System.out.println(x: "+-------------------------------------------------------+");
207                 goldShown = true;
208             }
209         }
210     }
211 }
212
```

Implementation

Source code in GitHub

ซอร์สโค้ดของโปรแกรมถูกจัดเก็บไว้บน GitHub เพื่อความสะดวกในการจัดการและตรวจสอบ

— **ชื่อ** Repository **ของกลุ่ม:** F2-256802-T01

— **โฟลเดอร์:** AS01-Rock-Paper-Scissors

How to compile and run the program

– **เปิด** Command Prompt **หรือ** Terminal
– **ไปยังโฟลเดอร์ที่เก็บไฟล์** Rock_Paper_Scissors.java
– **ใช้คำสั่ง** javac Rock_Paper_Scissors.java **เพื่อคอมไพล์โปรแกรม**
– **ใช้คำสั่ง** java Rock_Paper_Scissors **เพื่อรันโปรแกรม**