

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

Team No.7

6830300037	นายกอบชัย	บำรุงตน
6830300258	นายธนภัทร	ขันติกิจ
6830300312	นายธรา	นาทิพย์สกุล

**Submit to
Asst.Prof.Dr.Kullawadee Somboonviwat**

03603112-65 Programming Fundamentals

Table of Contents

a.Rock-Paper-Scissors Program

- i. Rules of the rock-paper-scissors game
- ii. Sample runs

b.Program Design

- i. Flowchart
- ii. Program Structure

c.Implementation

- i.Source code in GitHub
 - Team Repository name:
 - Folder: AS01-Rock-Paper-Scissors
- ii. How to compile and run the program

a. Rock-Paper-Scissors Program

i. Rules of the rock-paper-scissors game

Rock = Heavy Attack

Paper = Defense

Scissor = Light Attack

Defense > Light Attack > Heavy Attack > Defense

คำสั่งต่าง ๆ

- พิมพ์ !exit เพื่อออกเกม
- พิมพ์ !winscore เพื่อ check winscore
- พิมพ์ !BuyPotion เพื่อซื้อ Potion
- พิมพ์ !UsePotion เพื่อใช้ Potion

Potion สามารถ heal HP ของ player ได้ +1 โดยสามารถซื้อ potion ได้ 1 รอบต่อครั้ง โดย player สามารถนำแต้ม winscore ไปซื้อ potion ได้โดยมี ราคา 1 winscore ต่อ potion

ii. Sample runs

```
Welcome to the Game!
Enter your name: |
```

⇒ เมื่อเริ่มเกม ใส่ชื่อของคุณเพื่อไปต่อ

ii. Sample runs (ต่อ)

```
Hello, Player(1)!  
Let's play! (Type 'Defense', 'Light Attack', or 'Heavy Attack')  
Rules: Defense > Light, Light > Heavy, Heavy > Defense  
Initial HP: 5 | Total Rounds: 10  
Type  
**!exit** to quit,  
**!winscore** to check your wins  
**!UsePotion** to using potion and  
**!BuyPotion** to buy a potion(cap only 1 per buying)  
  
Match 1 (HP: 5/5)  
Player(1) => █
```

⇒ หลังจากใส่ชื่อมาแล้ว ตัวเกมจะทำการอธิบายระบบการเล่นและคำสั่งต่าง ๆ

```
Match 1 (HP: 5/5)  
Player(1) => Light Attack  
You win this match!  
Enemy choose: Heavy Attack  
Your HP is: 5 now.  
  
Match 2 (HP: 5/5)  
Player(1) => █
```

⇒ เมื่อผู้เล่นทำการพิมพ์คำสั่งโจมตี ตัวเกมจะทำการแสดงผลว่าผู้เล่นชนะหรือแพ้และจำทำการบอกด้วยว่า ศัตรูใช้ถ้าโจมตีอะไร (ถ้าผู้เล่นชนะ)

```
Match 2 (HP: 5/5)  
Player(1) => !winscore  
Matches Won: 1 out of 1 played.  
  
Match 2 (HP: 5/5)  
Player(1) => █
```

⇒ เมื่อผู้เล่นทำการพิมพ์คำสั่ง !winscore เพื่อเช็คจำนวนครั้งที่ชนะ

ii. Sample runs (ต่อ)

```
Match 2 (HP: 5/5)
Player(1) => !BuyPotion
You want to buy a potion? (Y/N)
Y
!!!!!! Buy potion success. !!!!
Current Potions: 1 | Current Wins: 0
```

→ เมื่อผู้เล่นพิมพ์คำสั่ง !BuyPotion ตัวเกมจะทำการถามผู้เล่นว่า ต้องการจะซื้อ Potion ไหม (YES/NO) โดยที่การซื้อ Potion คือ การนำ win ของผู้เล่นมาใช้ในการซื้อ และถ้าผู้เล่นทำการเลือก Y ระบบก็จะขึ้นคำสั่งว่าการซื้อสำเร็จ และบอกจำนวน Potion ที่เหลือและ win

```
Match 2 (HP: 5/5)
Player(1) => Light Attack
You lost this match!
Enemy choose: Defense
Your HP is: 4 now.
```

→ เมื่อผู้เล่นทำการพิมพ์คำสั่งโจมตี ตัวเกมจะทำการแสดงผลว่าผู้เล่นชนะหรือแพ้และจำทำการบอกด้วยว่า ศัตรูใช้โจมตีอะไร (ถ้าผู้เล่นแพ้)

```
Match 3 (HP: 4/5)
Player(1) => UsePotion
Invalid input!!!!
```

→ เมื่อผู้เล่นพิมพ์คำสั่งผิด

```
Match 3 (HP: 4/5)
Player(1) => !UsePotion
Used Potion! HP restored(+1). Current HP: 5
Potions left: 0

Match 3 (HP: 5/5)
Player(1) => █
```

⇒ เมื่อผู้เล่นทำการใช้คำสั่ง !UsePotion เลือกของผู้เล่นก็จะทำการเพิ่มขึ้นมา 1 หน่วย

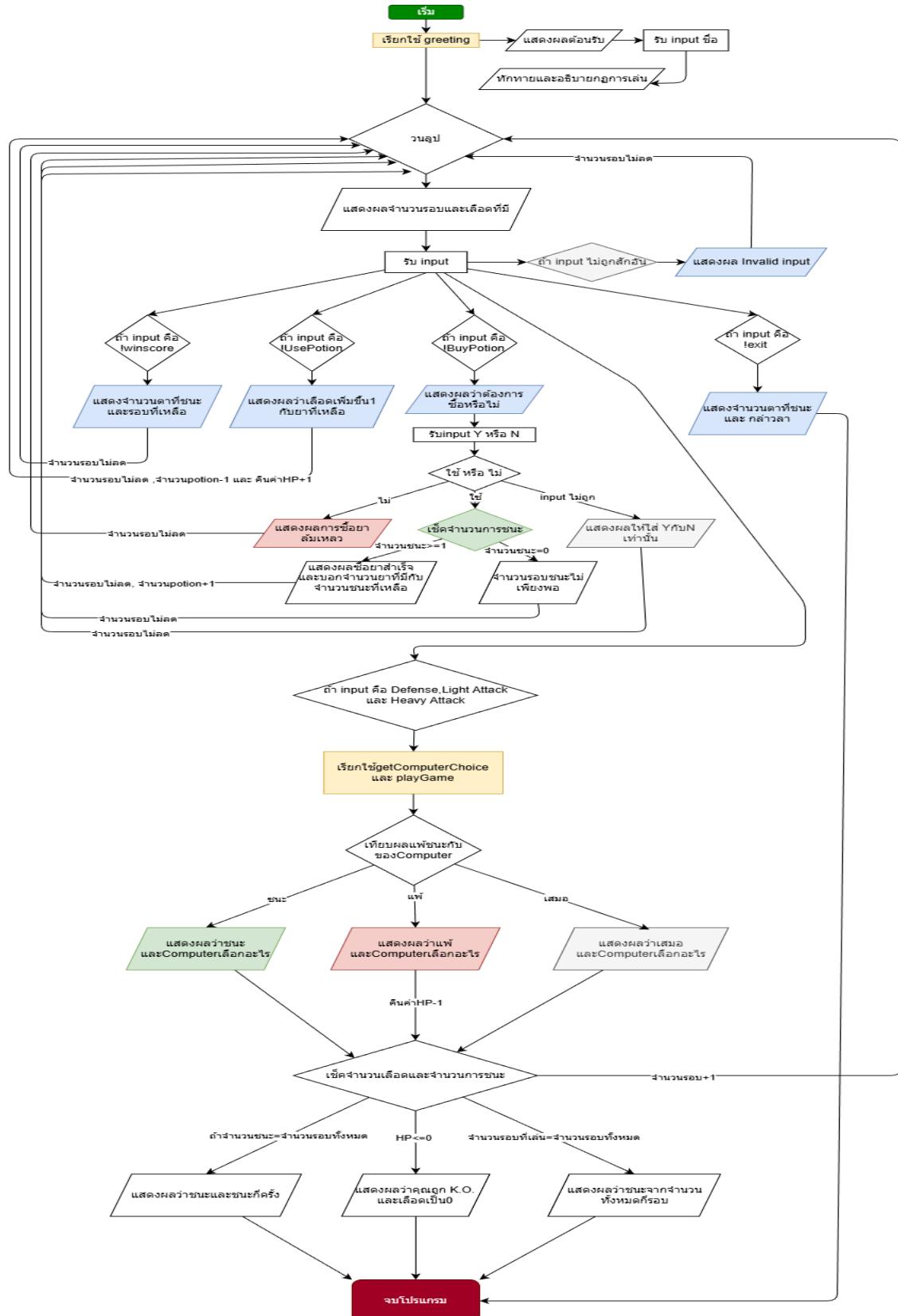
```
Match 3 (HP: 5/5)
Player(1) => !exit
Total Matches Won: **1/3**
Bye! See you! Player(1)

End Games
Final Win Score: 1
```

⇒ เมื่อผู้เล่นพิมพ์คำสั่ง !exit เพื่อออกเกม ตัวเกมจะแสดงจะแสดงว่าผู้เล่นชนะไปกี่ครั้งจากจำนวนการเล่นทั้งหมดของตนนั้น

b.Program Design

i.Flowchart



b.Program Design

ii. Program Structure

1. Global Constants & Variables โปรแกรมกำหนดค่าคงที่และตัวแปรสำคัญ เช่น

- ชนิดการโจมตี \Rightarrow Heavy Attack, Defense, Light Attack
- จำนวน HP เริ่มต้น \Rightarrow STARTING_HP = 5
- จำนวนรอบสูงสุดของเกม \Rightarrow MAX_ROUND = 10

2. Methods/Functions

2.1) greeting(Scanner sc,int winCount,int HP)

- แสดงข้อความต้อนรับ
- รับชื่อผู้เล่น
- อธิบายวิธีเล่นและคำสั่งพิเศษ

2.2) getComputerChoice()

- สุ่มการกระทำของศัตรู
- คืนค่า "Defense", "Light Attack" หรือ "Heavy Attack"

2.3) userWin(), userLose(), userDraw()

- แสดงผลลัพธ์ของแต่ละรอบ
- ใช้ภายในระบบตัดสินแพ้-ชนะ
- คืนค่าจำนวน win เพิ่มหรือไม่

2.4) HP System (yourHP_W, yourHP_L, lowHP)

- yourHP_W() \Rightarrow HPไม่ลดหรือเพิ่มหากผู้เล่นชนะหรือเสมอ
- yourHP_L() \Rightarrow ลด HP เมื่อแพ้(ค่าติด -1)
- lowHP() \Rightarrow เตือนเมื่อ HP ต่ำกว่า 2

ii. Program Structure (ต่อ)

2.5) playGame(playerChoice, computerChoice, currentHP)

- เปรียบเทียบผลแพ้-ชนะ
- คืนค่า {เพิ่ม win?, ปรับ HP?}
- แสดง HP ปัจจุบัน

3. Main Program (main method) โครงสร้างการทำงานของ main():

3.1) สร้าง Scanner

3.2) ตั้งค่า winCount, HP, roundCount

3.3) เรียก greeting() เพื่อเริ่มเกม

3.4) เข้าสู่ลูปเกม (while)

- แสดงรอบและ HP
- อ่าน input จากผู้เล่น
- ตรวจสอบคำสั่งพิเศษ:
 - !exit ⇒ เพื่อออกเกม
 - !winscore ⇒ ดูจำนวนคะแนนชนะ
 - !UsePotion ⇒ ใช้ยา
 - !BuyPotion ⇒ ซื้อยา

- ถ้าเป็นคำสั่งโจมตี (Defense, Light Attack, Heavy Attack)
 - สุ่มเลือกของศัตรู
 - เรียก playGame()
 - อัพเดต win และ HP
- ถ้าพิมพ์ผิด ⇒ แจ้ง Invalid input

3.5) เมื่อออกลูป:

- แสดง “End Games”
- แจ้งผลลัพธ์ว่าแพ้หรือชนะ HP/ชนะครบ 10 รอบ /หรือเล่นจนจบ

c.Implementation

i. Source code in GitHub

- Team Repository name:
AS01-Rock-Paper-Scissors-_Group7
- GitHub Repository URL:
https://github.com/Tara-Nawa/AS01-Rock-Paper-Scissors-_Group7
- Description:
 - FlowChart.drawio
 - DefenseLightHeavy.java

ii. How to compile and run the program

- เปิด Terminal หรือ Command Prompt ในโฟลเดอร์ที่คุณเก็บไฟล์ไว้ และพิมพ์คำสั่ง “javac SourceCode.java” เมื่อกด Enter และ พิมพ์ “java SourceCode” หลังจากนั้น เกมจะเริ่มทำงาน