Programming 1

Aryobarzan Atashpendar (aryobarzan.atashpendar@uni.lu)

Lab 4 – Iterative Structures

Slam Dunk © *Takehiko Inoue, Toei Animation*

Exercise 1 – Slam Dunk

Sakuragi from the *Slam Dunk* series wishes to practice his basketball throws by attempting them from different distances to the hoop.

1° Continuously ask the user to input a positive integer (excluding 0) in the terminal, which constitutes the distance in meters that Sakuragi is attempting his throw from. Print out the following message each time: 'Sakuragi attempts a **INPUT_NUMBER**m throw.'

2° Your program will keep track of the maximum, minimum and the sum of all the distances.

3° Once the user has input 0, you stop asking for input and print out the minimum, maximum and total sum. Additionally, you print out the average of all the distances.

Afterwards, Sakuragi's teammate Rukawa joins him for some practice as well, though Rukawa is more methodical in his throws:

- If the user has input at least 1 positive integer and the minimum value is not the same as the maximum value, and the maximum value is greater than 1, Rukawa will attempt throws starting from the minimum, with the distance always increasing by 1m until he has reached the maximum.
    - For each throw, print the following to the console: 'Rukawa manages his **CURRENT_DISTANCE**m throw.'
- For example, if the maximum is 6 and the minimum 2, Rukawa would attempt throws at 2m, 3m, 4m, 5m and 6m.
- Which loop structure did you end up using? Explain as a short comment under your code.

Persona © *Atlus*

Exercise 2 – I am thou, thou art I

In the *Persona* video game series, the player can perform different actions in combat. Create a game loop that continuously asks the user for an action to perform:

- If the user enters "attack", the game should simulate an attack by printing out "Attacking..."
- If the user enters "persona" or "change", the game randomly chooses a Persona to change to ("Anzu", "Pixie", "Oni") and prints out "Changing Persona to persona...", with 'persona' being replaced by the chosen Persona's name, e.g., Oni. Use a **switch expression** (hint below) to randomly choose a Persona.
- If the user enters "flee", the game loop ends with a final message "Fleeing..."
- If the user enters anything else, your program should ignore it by printing "Unknown command." and ask for another command.

To check which action to perform based on the user's input (attack, persona, change, flee), use a switch. You should have **no** if-conditions in your final implementation.

Random

To generate a random number, you can use the Random class that comes with native Java.

Switch expression

```java
int number = 2;
String name = switch(number) {
    case 0 -> "zero";
    case 1 -> "one";
    case 2 -> "two";
    default -> "unknown";
}
// In this case, name would be assigned "two".
```

Exercise 3 – Money money money

Write a program that calculates all possible combinations of 5, 10 and 20 euro bills which result in 100€. Print each combination, as well as the total number of possible combinations in the end.

Exercise 4 – Fourth time's the Fibocharm!

Ask the user for a positive multiple of 4 other than 0, denoted by n. If they enter an invalid number, keep asking until it's a valid number!

Afterwards, print out the Fibonacci sequence $\mathscr{F}$ until the n$^{\text{th}}$ term $\mathscr{F}_n$.

$$\begin{cases} \mathscr{F}_0 = 0 \\ \mathscr{F}_1 = 1 \\ \mathscr{F}_n = \mathscr{F}_{n-1} + \mathscr{F}_{n-2} \quad n \geq 2 \end{cases}$$