

Groupe 1

Bloc 1 TP 4

1. Analyse du problème	2
2. La conception de la solution	2
3. Réalisation	2
4. Test	4
5. Retour d'expérience	4

1. Analyse du problème

Le but est de réaliser un Shifumi en Java (le compte-rendu ne parle que des deux premières étapes).

2. La conception de la solution

La règle du jeu consiste à choisir entre une pierre, une feuille et un ciseau. La pierre écrase le ciseau, la feuille recouvre la pierre et le ciseau coupe la feuille.

Il faut simuler ce jeu sur ordinateur, avec le programme qui choisira aléatoirement l'un des trois objets.

3. Réalisation

D'abord, on demande le nombre de rounds (3, 5 ou 10).

On récupère ce que l'utilisateur écrit. Si la valeur saisie n'est ni 3, ni 5, ni 10, on redemande avec une boucle "while" jusqu'à ce que la valeur soit correcte. Un message spécifique est ajouté avec "if" pour indiquer que l'utilisateur a entré une mauvaise ou bonne valeur.

```
import java.util.Scanner;

/*
 * La classe principale pour le jeu Shifumi
 * Ce programme va simuler un jeu contre l'ordinateur avec des règles classiques.
 * Règles:
 * - La pierre bat la pierre
 * - La pierre bat le ciseau
 * - La pierre bat le papier
 * - Egalité si pareil
 */
public class Shifumi {
    public static void main(String[] args) {
        // Déclaration des variables
        int round = 0;
        char playerItem = '0';
        char botItem = '0';
        Scanner sc = new Scanner(System.in);

        // Etape 1: Afficher et demander le nombre de points pour la partie
        while(round != 3 && round != 5 && round != 10) {
            System.out.println("Combien souhaitez-vous de rounds ? (3, 5 ou 10)");
            round = sc.nextInt();

            if (round != 3 && round != 5 && round != 10) {
                System.out.println("La valeur entrée est incorrect.");
            } else
                System.out.println("Vous avez rentré : " + round);
        }

        // Etape 2: Afficher et demander l'item
        while(playerItem != 'P' && playerItem != 'F' && playerItem != 'C') {
            System.out.println("Veuillez choisir entre Pierre (P), Feuille (F) ou Ciseau (C).");

            playerItem = sc.next().charAt(0);

            if (playerItem != 'P' && playerItem != 'F' && playerItem != 'C') {
                System.out.println("L'objet entré est incorrect.");
            } else
                System.out.println("Vous avez rentré : " + playerItem);
        }
    }
}
```

I

2

Ensute, on demande à l'utilisateur quel objet il souhaite utiliser.

On lui demande d'écrire le mot ou simplement la lettre correspondant à l'objet.

On utilise "sc.next().charAt(0);" pour récupérer la première lettre de ce que l'utilisateur a entré.

De la même manière, une boucle "while" vérifie si la valeur correspond à 'P', 'F' ou 'C'. Sinon, la demande est répétée jusqu'à ce qu'elle soit correcte. Une vérification avec "if" permet aussi d'afficher un message personnalisé.

```
Shitumi.java
29     while(round != 3 && round != 5 && round != 10) {
30         System.out.println("Combien souhaitez-vous de rounds ? (3, 5 ou 10)");
31
32         round = sc.nextInt();
33
34         if (round != 3 && round != 5 && round != 10) {
35             System.out.println("La valeur entrée est incorrect.");
36
37         } else
38             System.out.println("Vous avez rentrez : " + round);
39     }
40
41 // Etape 2: Afficher et demander l'item
42 while(playerItem != 'P' && playerItem != 'F' && playerItem != 'C') {
43     System.out.println("Veuillez choisir entre Pierre (P), Feuille (F) ou Ciseau (C).");
44
45     playerItem = sc.next().charAt(0);
46
47     if (playerItem != 'P' && playerItem != 'F' && playerItem != 'C') {
48         System.out.println("L'objet entrée est incorrect.");
49
50     } else
51         System.out.println("Vous avez rentrez : " + playerItem);
52 }
53
54 //Etape 3
55 int colonne = (int) (Math.random())*(3)+1;
56
57 switch(colonne) {
58
59     case 1:
60         botItem = 'P';
61         break;
62
63     case 2:
64         botItem = 'F';
65         break;
66
67     default:
68         botItem = 'C';
69         break;
70
71 }
72
73 System.out.println("L'ordinateur a choisié : " + botItem);
74
75
76
77
78 }
```

4. Test

Nous avons testé chaque ligne et étape du code avant de passer à la suite, afin de pouvoir identifier les erreurs dès le début.

5. Retour d'expérience

C'est notre premier programme concret en Java (une vraie interface utilisateur avec un objectif).