

B1 TP2 Compte Rendu
Shifumi JAVA

Sommaire :

Analyse du problème

Conception de la solution

Réalisation

Test

Retour expérience

Quelques remarques

Analyse du problème :

Lors de ce td nous devons configurer un code JAVA afin de pouvoir effectuer un "shifumi" contre l'ordinateur. Nous aurons la possibilité de choisir le nombre de manches et de choisir nos coups.

Conception de la solution :

Premièrement nous ouvrons eclipse et créons un projet et une classe puis commençons à coder :

```
package shifumiordi;

import java.util.Scanner;

/**
 * La classe principale pour le jeu Shifumi (pierre-feuille-ciseaux).
 * Ce programme ajoute un jeu contre l'ordinateur avec les règles classiques :
 * - le papier bat la pierre
 * - les ciseaux battent le papier
 * - la pierre bat les ciseaux
 * - Si deux éléments similaires : égalité
 *
 * @author come
 * @version 1.0
 */

public class Shifumipc {

    // variables
    int nbPoints;
    int aleatoire;
    char choixJoueur;
    char choixOrdi;
    int ptOrdi;
    int ptJoueur;
    int round;

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Shifumipc jeu = new Shifumipc();
        jeu.demarrer(sc);
    }

    public void demarrer(Scanner sc) {
        // Étape 1 : définir le nombre de points
        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 incorrecte.");
            } else {
                System.out.println("Vous avez choisi : " + round + " rounds.");
            }
        }
    }
}
```

Nous insérons d'abord des commentaires dans lesquels nous dictons les règles du jeu puis nous définissons les variables (entier/chaine de caractère...)

Réalisation :

Lors des 4 première étapes:

```
public void demarrer(Scanner sc) {
    // Étape 1 : définir le nombre de points
    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 incorrecte.");
        } else {
            System.out.println("Vous avez choisi : " + round + " rounds.");
        }
    }

    // Étape 6 : Boucle des manches
    for (int i = 1; i <= round; i++) {
        System.out.println("Manche " + i);

        // Étape 2 : Choix du joueur
        do {
            if (avecPuits) {
                System.out.println("\nChoisissez un outil : P (pierre), F (feuille), C (ciseaux), Q (puits)");
            } else {
                System.out.println("\nChoisissez un outil : P (pierre), F (feuille), C (ciseaux)");
            }
            choixJoueur = sc.next().toUpperCase().charAt(0);
            System.out.println("Vous avez choisi : " + symbole(choixJoueur));
        } while (choixJoueur != 'P' && choixJoueur != 'F' && choixJoueur != 'C' && (!avecPuits || choixJoueur != 'Q'));

        // Étape 3 : Choix aléatoire de l'ordinateur
        if (avecPuits) {
            aleatoire = (int)(Math.random() * 4) + 1; //option pour parties avec Puits
        } else {
            aleatoire = (int)(Math.random() * 3) + 1;
        }

        if (aleatoire == 1) choixOrdi = 'P';
        else if (aleatoire == 2) choixOrdi = 'F';
        else if (aleatoire == 3) choixOrdi = 'C';
        else choixOrdi = 'Q';

        // Étape 4 : Révéler le choix après attente
        System.out.println("L'ordinateur réfléchit...");
        try {
            Thread.sleep(3000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("L'ordinateur a choisi : " + symbole(choixOrdi));
    }
}
```

Ces 4 premières étapes ont été effectuées en parallèle de notre enseignante nous avons utilisé des conditions telles que for, if/else if/else, while, do pour faire des boucles et permettre de simplifier les étapes. Lors de l'étape 2 on demande à l'utilisateur de choisir entre pierre, feuilles et ciseaux, nous reprenons la première lettres saisie "P, F ou C" pour définir la variable. Nous avons inséré la lettre Q pour le puits car le P est déjà pris. L'étape 3 consiste à déterminer le choix. L'étape 4 attend 3 secondes avant d'afficher le résultat. L'étape 6 consiste a faire une boucle lorsque la manche n'est pas terminé, quand la manche est terminée il prend la valeur 1 à chaque point.

Lors des dernières étapes :

```
// Étape 5 : Déterminer le gagnant
if (choixOrdi == choixJoueur) {
    System.out.println("Égalité !");
} else if ((choixOrdi == 'P' && (choixJoueur == 'C')) ||
           (choixOrdi == 'C' && (choixJoueur == 'F')) ||
           (choixOrdi == 'F' && (choixJoueur == 'P')) ||
           (choixOrdi == 'P' && choixJoueur == 'Q') ||
           (choixOrdi == 'C' && choixJoueur == 'Q') ||
           (choixOrdi == 'Q' && choixJoueur == 'F')) {
    ptOrdi++;
    System.out.println("L'ordinateur gagne la manche.");
} else {
    ptJoueur++;
    System.out.println("Vous gagnez la manche !");
}

// Étape 7 : Fin de partie
System.out.println("Résultat final");
System.out.println("Vous : " + ptJoueur + " | Ordinateur : " + ptOrdi);

if (ptJoueur > ptOrdi) {
    System.out.println("Vous gagnez la partie !");
} else if (ptJoueur < ptOrdi) {
    System.out.println("L'ordinateur a gagné la partie !");
} else {
    System.out.println("Égalité");
}

// Demander si l'utilisateur veut rejouer
System.out.println("Voulez-vous rejouer ? (O/N)");
char rejouer = sc.next().toUpperCase().charAt(0);
if (rejouer == 'O') {
    ptJoueur = 0;
    ptOrdi = 0;
    round = 0;
    demarrer(sc);
} else {
    System.out.println("Merci d'avoir joué !");
}

// Étape 8 : Fonction utilitaire pour afficher les symboles
public String symbole(char choix) {
    switch (choix) {
        case 'P': return "■";
        case 'F': return "□";
        case 'C': return ">";
        case 'Q': return "O"; // symbole pour Puits
        default: return "erreur";
    }
}
```

Nous avons effectué une condition lors de l'étape 5 Si le choix de l'ordinateur est le même que celui du joueur, la manche est nulle. Puis le Sinon Si renvoie vers les règles du jeu (qui gagne qui) pour déterminer le message d'affichage gagnant de l'ordinateur. En revanche le Sinon renvoie vers toutes les alternatives gagnantes du joueur. Étape 7: Nous affichons les résultats de fin de manches, si le joueur a plus de points que l'ordi il gagne si l'inverse il perd, sinon c'est une égalité et la manche continue. Étape 8 : On demande si l'utilisateur veut rejouer (erreur emplacement etape 8 code) en utilisant le même principe que la variable prenant l'entrée du choix utilisateur avec le scanner vérifiant l'entrée du clavier si il tape un mot en O (oui) la partie recommence en redémarrant la fonction demarrer.sc défini étape 6 .Sinon s'il tape N (non) la partie s'arrête. Et enfin étape 9 on signe des symboles au choix grâce à la variable switch case vu dans le cours, en indiquant la valeur par défaut à une erreur.

Test :

Lors de test nous n'avons pas eu d'erreurs de code dans la première séance. Cependant lors de la seconde séance nous avons eu recours à plusieurs erreurs comme l'oubli de déclaration de variables. Nous avons recherché des informations sur des codes JAVA sur internet pour réaliser certaines étapes et trouver la condition/variable adaptée. Par exemple

Retour expérience :

Je trouve que le code du td est trop long on se perd au fur et à mesure du travail, de plus mon travail ayant été enregistré sur le pc de la salle F205 a été supprimé, j'ai donc retrouvé mon projet JAVA vide et donc perdu du temps sur mon travail et j'ai eu des difficultés à retrouver un code complet avec la structure du cours précédent.

Quelques remarques :

Ayant retrouvé mon travail supprimé et n'ayant pas été le seul, j'ai été déçu et cela a impliqué ma discipline lors de la réalisation de la seconde séance de cours.

