

MATA04040100 Antoine MATHIEU

MILV09060100 Vincent MILLOUR

SUJET 2 : Programmation par aspect

Nous avons réalisé le sujet de la programmation par aspect en créant deux aspects. Le premier s'occupe de créer un fichier log qui enregistre les mouvements réalisés par chaque joueur tandis que le second ajoute les vérifications permettant de savoir si le mouvement d'une pièce réalisé par un joueur est autorisé.

Fichier de log :

```
pointcut isMakeMove(Player player, Move mv) : call(void *.movePiece(Move))

//after pour enregistrer les coups dans un fichier
after(Player player, Move mv) : isMakeMove (player,mv) {
    logMove(player.toString()+": " + mv.toString() + "\n");
}

// Méthode pour enregistrer les coups dans un fichier
private void logMove(String move) {
    try {
        FileWriter fileWriter = new FileWriter("chess_moves.log", true);
        BufferedWriter bufferedWriter = new BufferedWriter(fileWriter);
        bufferedWriter.write(move);
        bufferedWriter.newLine();
        bufferedWriter.close();

    } catch (IOException e) {

    }
}
```

Ici, le pointcut capture tous les appels de la méthode movePiece permettant de déplacer les pièces sur le plateau.

On récupère en argument player et mv (mouvement) pour afficher dans le fichier de log quel joueur a joué quelle pièce.

Fichier de validation de mouvements :

```
pointcut validationMovement(Player player, Move mv) : call(boolean agent.Player.makeMove(Move))

boolean around(Player player, Move mv) : validationMovement(player, mv) {

    //On regarde si la position initial est à l'intérieur du plateau
    if (mv.xI < 0 || mv.xI >= Board.SIZE || mv.yI < 0 || mv.yI >= Board.SIZE)
    {
        System.out.println("La position initial n'est pas dans la plateau");
        return false;
    }

    //On regarde si la position finale est à l'intérieur du plateau
    if (mv.xF < 0 || mv.xF >= Board.SIZE || mv.yF < 0 || mv.yF >= Board.SIZE) {
        System.out.println("La position finale n'est pas dans la plateau");
        return false;
    }

    // on regarde si il y a une pièce sur l'emplacement choisi
    if (!player.getPlayground().getGrid()[mv.xI][mv.yI].isOccupied())
    {
        System.out.println("Il n'y a pas de pièces sur la case de départ choisie");
        return false;
    }
}
```

Ici, le pointcut capture la méthode makeMove des agents (IA et humain) et utilise un around pour ajouter des conditions à cette méthode. On récupère en argument player ainsi que mv (mouvement) pour vérifier plusieurs conditions listées ci-dessous :

- La position initiale/finale se situe sur le plateau
- Il y a une pièce appartenant au joueur sur la case initiale
- Il n'y a pas de pièces du joueur sur la case finale
- La pièce déplacée n'en traverse pas d'autres durant son déplacement

SUJET 4 : Python

Polymorphisme : Pour montrer le Polymorphisme en Python, nous avons créé une classe mère *Forme* avec trois classes filles *Carre*, *Triangle* et *Cercle*. La classe *Forme* possède une méthode *Calculer_Aire* qui sera redéfinie par chacune des classes filles. Ce type de polymorphisme est appelé l'overriding.

L'overloading, quant à lui, n'est pas pris en charge par Python. En effet, même s'il est possible de définir plusieurs méthodes avec un nom identique, seule la dernière à avoir été définie sera prise en compte.

Généricité : Pour montrer la Généricité en Python, nous avons créé une classe *Generique* qui affiche un contenu générique. Cette classe est générique car elle peut prendre en argument des entiers, des chaînes de caractères, des floats et autres.

Modularité : Pour montrer la Modularité en Python, nous avons créé une petite calculatrice où les calculs sont gérés dans un module appelé *calculateur* et l'affichage dans un module appelé *affichage*.

Ces exemples montrent différents types de modularité en mettant des classes et des méthodes en module ou encore avec l'importation de modules déjà réalisés comme le module *math*.

Analyse de données : Pour montrer l'Analyse de données en Python, nous avons utilisé la librairie *pandas* pour lire un fichier *csv* et garder les patients avec le label 4.