

TD / TP n°3

Objectifs:

Comprendre l'asynchronicité, au travers de la programmation événementielle et des exceptions.

Préparer son programme à être interrompu.

Comprendre la différence entre une erreur et une exception.

Notions à connaître:

Exceptions, callbacks, passage de paramètres, et appels synchrones/asynchrones

Exercice 1: Exceptions / Erreurs

En Java, une erreur est signalée par une exception : pointeur nul, élément en dehors du tableau, division par 0, etc...

- *Ecrire une méthode calculant la valeur de Fibonacci associée à un int. Cette méthode lèvera l'exception `IndexOutOfBoundsException` si l'int passé en paramètre est négatif.*
- *Ecrire une méthode de test calculant et affichant la valeur de Fibonacci associée à -137 et 137.*

Exercice 2: Exceptions / Information

On souhaite faire un algorithme de parcours de labyrinthe. Un personnage se déplace linéairement, et s'arrête lorsqu'il rencontre un obstacle.

Une façon de faire consiste à décomposer l'action en plusieurs étapes:

- voir si on peut aller vers le {haut;bas;gauche;droite}
- si on peut, avancer dans cette direction
- si on ne peut pas, choisir une autre direction et recommencer

Le mécanisme des exceptions permet de faire les deux en une seule passe:

- essayer de se déplacer dans un sens
- si exception, changer de direction

Le système d'exceptions garantit que si le mouvement est impossible, l'action sera suspendue.

- *Déclarer la classe `Personnage`, qui a une méthode `public void allerACase(int x, int y) throws MouvementIllegalException, ObstacleException`. On ne s'occupera pas de l'implémentation pour l'instant.*
- *Quelle est la différence entre un mouvement illégal et un obstacle? Pourquoi a t'on besoin des deux?*
- *Inventer un algorithme exploitant cette méthode permettant de se promener dans le labyrinthe.*
- *Ecrire une classe `Promenade` qui exploite cet algorithme pour aller d'un point A à un point B:*
`public void trouverChemin(int ax, int ay, int bx, int by)`

Exercice 3 : Programmation événementielle

Reprenons l'exemple précédent. Au lieu d'exceptions, on pourrait utiliser un système d'évènements. Réfléchir à une façon de modifier l'algorithme pour prendre en compte cette façon de faire.

Quelles sont les différences majeures entre ces deux algorithmes? (indice : même différence qu'entre une boucle while et une boucle for).

A préparer pour le TP correspondant:

Nettoyer le pseudo-code en vue d'une implémentation sur machine. La classe Personnage sera fournie, ainsi qu'une classe Labyrinthe dans laquelle le Personnage se promènera.

A retenir pour le TD suivant:

Exceptions, asynchronicité.