

### TP : Monde des ROBOTS

Dans un environnement virtuel évoluent des robots virtuels. Il existe plusieurs types de robots : des robots pollueurs et des robots nettoyeurs

Les robots pollueurs se baladent dans le monde virtuel et ils déposent des déchets sur chacune des cases qu'ils parcourent :

- Des robots pollueurs qui vont tout droit
- Des robots pollueurs sauteurs
- Des robots pollueurs qui se déplacent librement

Les robots nettoyeurs, quant à eux, ne supportent pas la vue de déchets et l'enlèvent dès qu'ils passent sur une case sale. Il existe trois sortes de robots nettoyeurs :

- Des robots nettoyeurs qui parcourent l'ensemble des cases du monde pour le nettoyer
- Des robots nettoyeurs qui ne nettoient qu'un déchet sur deux.
- Des robots nettoyeurs qui parcourent librement dans toutes les directions

Le but de ce TP est développer une application qui gère les déplacements des robots via une interface graphique Java en programmant une version personnalisée en utilisant un IDE (Eclipse , NetBeans....) .

Pour les interfaces graphiques, vous pouvez utiliser les deux bibliothèques SWING, AWT...

Voici qlq lignes pour vous guider à la conception de ce projet.

Les robots pollueurs et nettoyeurs se déplacent sur une grille représentée par une matrice booléenne. La case est false si la case est propre et elle vaut true si elle contient un papier gras.

### La classe Monde

Écrire la classe Monde qui contient les variables, le constructeur et les méthodes suivantes :

- Le nombre de lignes nbL, le nombre de colonnes nbC, et une matrice booléenne Mat de nbL lignes et nbC colonnes.
- Ecrire un constructeur qui crée un monde ( par exemple 10 X10) sans papiers gras.
- Ecrire une méthode qui met un papier gras dans la case (i, j). Si (i,j) n'est pas dans la matrice Mat une Exception sera lancée.
- Ecrire une méthode qui prend un papier gras en (i, j). Si (i,j) n'est pas dans la matrice Mat une Exception sera lancée.
- Ecrire une méthode qui teste si la case (i,j) a un papier gras. Si (i,j) n'est pas dans la matrice Mat une Exception sera lancée.
- Ecrire une méthode qui compte et renvoie le nombre de papiers gars dans le monde.
- Ecrire une matrice qui permet d'afficher la matrice Mat.

### **La classe Robot**

Ecrire une classe Robot abstraite qui contient les champs et méthodes suivantes :

- la position posx, posy du robot, un Monde m car il faut que le robot connaisse le monde pour pouvoir s'y déplacer et agir.
- Écrire un constructeur **Robot(int x, int y)** qui crée un robot a la position (x,y).
- Écrire un constructeur qui crée un robot avec une position aléatoire.
- Écrire une méthode pour se déplacer en (i,j). Si (i,j) n'est pas dans la matrice Mat une Exception sera lancée.
- Écrire une méthode abstraite parcourir() qui lance une Exception.

### **La classe RobotpollueurToutDroit**

Un **RobotpollueurToutDroit** est un Robot qui peut mettre un papier gras là où il se trouve. Il a un numéro, le numéro de la colonne où il va peut se rendre pour commencer sa mission. Le **PollueurToutDroit** se positionne d'abord dans sa colonne de départ sur la case (0, ColDepart), puis il va tout droit vers le sud en visitant chaque case de la colonne et dépose un papier gras sur chacune d'elle. Il s'arrête dans la dernière case de la colonne.

Écrire la classe **PollueurToutDroit** : ses attributs, ses constructeurs et ses méthodes.

### **La classe RobotpollueurSauteurs**

Écrire cette classe avec ses attributs, ses constructeurs et ses méthodes.

### **La classe Robotpollueurlibre**

Écrire cette classe avec ses attributs, ses constructeurs et ses méthodes.

### **La classe RobotNettoyeur**

Un **Robotnettoyeur** est un Robot qui parcourt tout le Monde pour le nettoyer.  
Écrire la classe Robotnettoyeur avec les méthodes nécessaires.

### **La classe RobotNettoyeur standard**

Écrire cette classe avec ses attributs, ses constructeurs et ses méthodes.

### **La classe RobotNettoyeur smart**

Écrire cette classe avec ses attributs, ses constructeurs et ses méthodes.

### **La classe RobotNettoyeur libre**

Écrire cette classe avec ses attributs, ses constructeurs et ses méthodes.

Voici un exemple d'une interface graphique d'un environnement 2D avec des robots.....

Nettoyer

Polluer

Réinitialiser le monde

Quitter

Nombre de papiers gras: 0

POLLUER droit

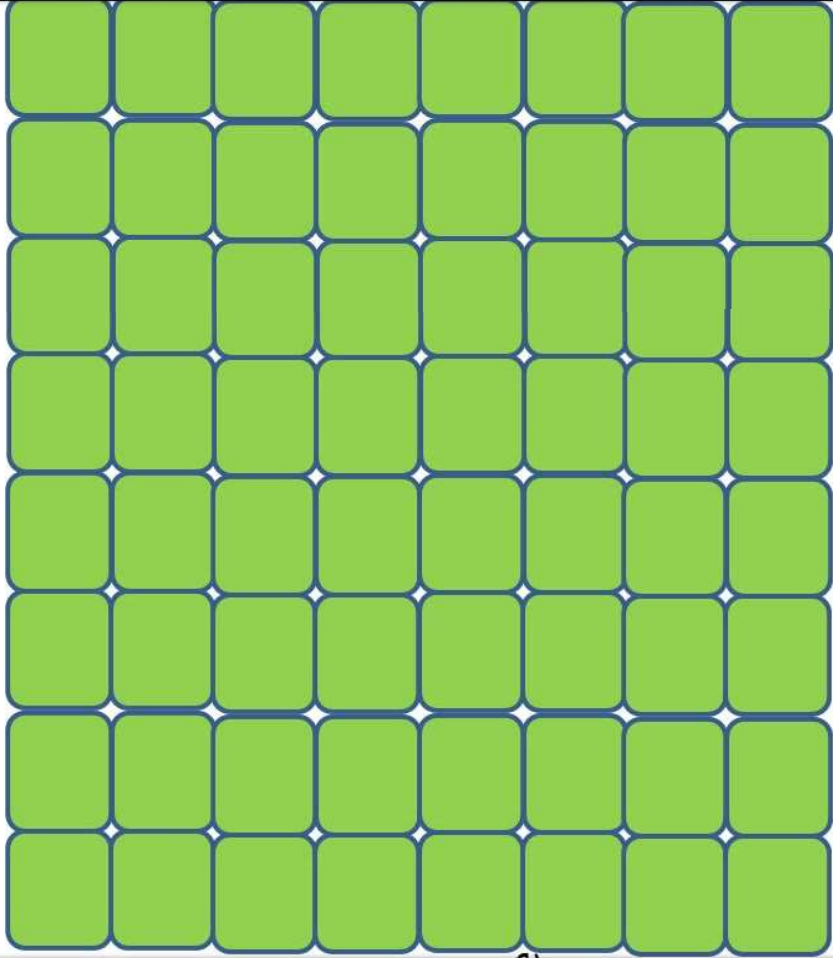
POLLUER Sauteurs



Haut  
Gauche  
Droite  
Bas

ROBOTS LIBRES

Quitter

MONDE



 Papier Gras  Case nettoyée

ROBOTS DROITS

Nettoyer

Haut  
Gauche  
Droite  
Bas

Smart robotic cleaner

Quitter

Nettoyer

Polluer

Réinitialiser le monde


Quitter

Nombre de papiers gras: 9

[illegible]

POLLUER droit

POLLUER  
Sauteurs



**Bas**

ROBOTS LIBRES

Quitter

ROBOTS DROITS

Nettoyer

**Bas**

Smart robotic cleaner

Quitter

