

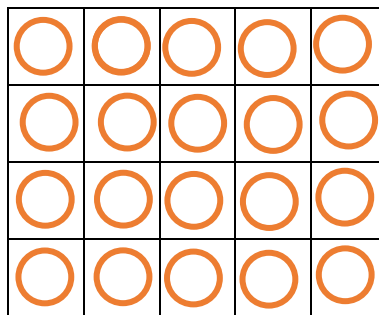
## Mini Projet :

### Stratégie de déplacement d'un robot

Nous allons étudier des stratégies de pilotage d'un robot ayant 3 fonctions :

- avancer,
- tourner,
- déposer une goutte au point où il se trouve.

L'objectif est de déposer une goutte dans chaque case d'une grille en le moins de temps possible. Pour cela vous allez définir une stratégie permettant de passer par tous les points de la grille, puis simuler les trajets afin de calculer le temps nécessaire.



*Exemple de disposition des gouttes sur une grille*

#### Rendu du Projet :

Date de rendu : 14 octobre

Envoyer à l'adresse mail : [clement.roch@ens-paris-saclay.fr](mailto:clement.roch@ens-paris-saclay.fr)

Les codes seront évalués suivant plusieurs critères représentant les compétences clés du cours. Une attention particulière devra être portée sur le fonctionnement du code, sa structure et sa lisibilité.

## Prise en main de la simulation du robot

Le code dans le script « rob\_follow\_instr.py » donne les fonctions de base du robot qui sont : initialiser la simulation, déplacer le robot, déposer et arrêter la simulation. Dans cette partie l'objectif est de déplacer le robot pour atteindre une succession de points.

### Question 1

Initialiser le robot en position (0,0) puis appeler la fin de la simulation. Votre code sera à la suite des fonctions prédéfinies.

### Question 2

Initialiser deux variables x1 et y1 valant 100 et 300 respectivement.

### Question 3

Déplacer le robot pour atteindre le point de coordonnées (x1, y1). Quel est le temps nécessaire pour effectuer cette action ?

### Question 4

Créer une nouvelle variable et modifier le code pour que le robot dépose au point si la variable vaut 1 et ne dépose pas si elle vaut 0.

### Question 5

Créer une fonction prenant en entrée une valeur x, une valeur y et un entier d qui :

- Déplace le robot au point de coordonnées (x,y)
- Dépose si d vaut 1
- Renvoie le temps total de ces opérations

On cherche maintenant à suivre une succession de points. Le tableau suivant présente la liste des points à atteindre et des points de dépose. La première ligne contient les coordonnées en x, la seconde les coordonnées en y et la troisième indique s'il s'agit du point de départ (-1), d'un point de passage (0) ou d'un point de dépose (1).

-150	-50	-30	0	50	130
-150	-50	30	0	200	20
-1	0	0	1	1	0

### Question 6

Créer trois listes représentant les trois lignes du tableau.

### Question 7

Ajouter les instructions pour que le robot se déplace en chaque point et effectue l'action demandée.

### Question 8

Créer une nouvelle liste qui contiendra autant d'éléments qu'il y a de points et qui stocke le temps nécessaire à chaque action.

### Question 9

Déterminer le temps total nécessaire pour effectuer ces opérations.

## Stratégie de remplissage

On s'intéresse maintenant à établir une stratégie pour remplir la grille. Grâce à cette stratégie les coordonnées des points à atteindre pourront être définies.

La grille à remplir sera représentée dans un premier temps comme une matrice de  $n$  lignes et  $m$  colonnes remplie de 0. L'action de remplissage sera représentée par le remplacement d'un 0 par la valeur 1.

### Question 10

Créez un nouveau script et initialisez deux variables  $n$  et  $m$  correspondant au nombre de ligne et de colonnes. (exemple  $n=3$ ,  $m=4$ )

### Question 11

Initialisez une matrice (type *numpy.array*) remplie de 0 de taille  $n*m$ .

### Question 12

Choisissez une stratégie de déplacement pour passer en tout point de la matrice. À l'aide d'un papier et d'un crayon, dessinez cette stratégie et exprimez les suites d'étapes nécessaires (exemple, avancer d'une ligne jusqu'à la fin puis avancer d'une case)

### Question 13

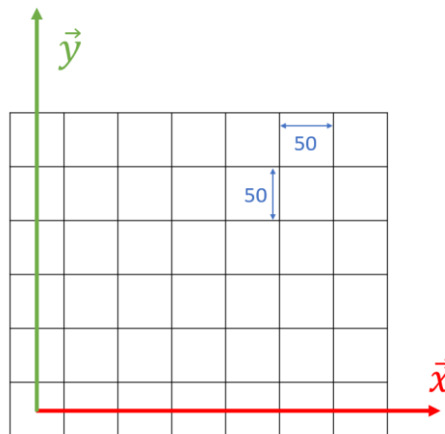
En vous aidant de l'étape précédente, ajoutez à votre code une succession d'étapes permettant de remplir la matrice avec des 1 dans l'ordre souhaité.

### Question 14

Testez votre code avec plusieurs tailles de matrices et vérifiez son bon fonctionnement. Une stratégie correctement mise en œuvre devrait donner une matrice ne contenant que des 1.

## Génération des trajectoires

La grille a une dimension physique donnée. Par exemple, une case d'une grille classique a une taille de 50 mm par 50 mm. On fixe l'origine des coordonnées au centre d'une case à un coin de la grille, suivant la convention suivante :



### Question 15

Exprimez les coordonnées de chaque centre de case en fonction des indices de la matrice. La convention des indices est libre (case [0,0] en haut-gauche, bas-gauche, bas-droit ou haut-droit).

### Question 16

Modifiez votre algorithme de remplissage pour calculer en chaque point les coordonnées correspondantes.

### Question 17

Générer la liste des coordonnées des points à atteindre et la liste des actions à effectuer sur le modèle du tableau utilisé précédemment.

## Simulation des trajectoires

On considère un point de départ en -150 ;-150

### Question 18

Simuler la trajectoire du robot pour atteindre les points définis par votre stratégie.  
Calculer le temps total de dépose.

### Question 19

Proposer une modification du code pour que la matrice utilisée de la question 10 à 14 ne soit pas remplie de 1 mais que chaque case contienne le temps auquel le robot passe par celle-ci.

## Import des données d'entrées

Les données d'entrées telles que la taille de la grille, la taille des cases et le point de départ sont donnés dans des fichiers textes (format donné dans le dossier « grilles »)

### Question 10

Modifiez votre script pour lire un fichier d'entrée et utilisez les données pour définir le problème.

### Question 11

Utilisez votre script pour générer les fichiers de trajectoires correspondant à l'application de votre stratégie sur les grilles prédéfinies.

## Export des trajectoires

Les coordonnées des points peuvent être transmises par l'intermédiaire d'un fichier texte sous le formalisme suivant (avec les points du tableau question 6) :

```
Fichier de dépose
Depart (-150,-150)
Points :
Passage (-50,-50)
Passage (-30,-30)
Depose (0,0)
Depose (50,200)
Passage (130,20)
Fin
```

Les points nommés « passage » correspondent à des points où le robot doit se rendre mais sans commettre d'action, et ceux nommés « dépose » correspondent aux points où le robot doit effectuer l'action de dépose.

Le point de départ sera donné en -300 ; -300

Il est possible que votre fichier ne contienne pas de points de passage.

#### Question 8

Modifiez votre script pour créer un fichier texte contenant les lignes suivantes :

```
Fichier de dépose
```

```
Depart (-300, -300)
```

```
Points :
```

*Note : un code exemple pour lire un texte est donné dans le dossier « exemple\_lecture »*

#### Question 9

Ajouter après « Points : » les coordonnées des points de passage et dépôt correspondant à votre stratégie puis finissez par « Fin »