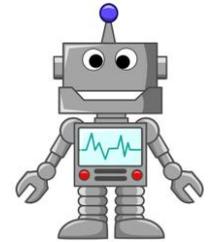
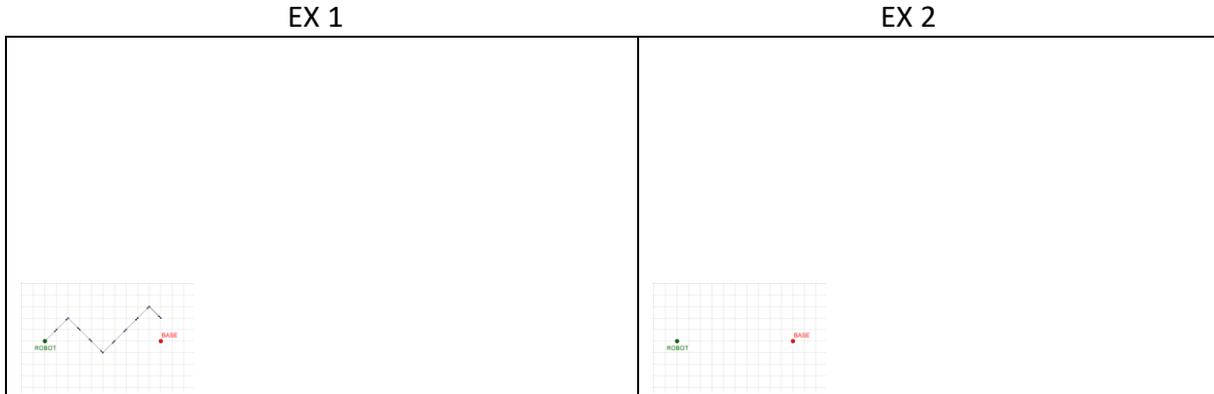


La marche du petit robot



Un petit robot doit rentrer à sa base. Il fonctionne de façon aléatoire en diagonale vers la droite ou vers la gauche avec la même probabilité.
On suppose que le robot fait exactement 10 déplacements et que la distance parcourue à chaque déplacement est identique.

Voici un exemple (EX 1) où le robot n'atteint pas sa base.
Réaliser un exemple où il réussit à rentrer (EX2).



On a programmé une telle marche avec le langage « python »

```

4
5 D=0
6 G=0
7 for i in range(10):
8     P=randint(0,1)
9     if P==0:
10        D=D+1
11    else:
12        G=G+1
13 print("D=",D,"G=",G)
14
    
```

Randint(0,1) génère un nombre entier aléatoire 0 ou 1

Première partie

- Expliquer comment est simulé le déplacement du robot ? En particulier expliquer le rôle de la variable P

.....

- Que représentent les variables G et D ? Quelles valeurs peuvent-elles prendre ?

.....

- Exécuter le programme : Qu'obtenez-vous ?

- Que pourrait-on ajouter dans le script pour avoir un chemin détaillé du petit robot du type 0001010110 ou DDDGDGDDG ?

.....

Tester le programme modifié, donner le résultat et tracer le chemin du robot

<pre> D= G =..... </pre>	
------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------

- On suppose qu’une fois le programme exécuté, la variable D a la valeur 5. Interpréter ce résultat.

.....
 En déduire une instruction à ajouter pour contrôler si le robot est rentré à sa base

Compléter le programme ci-dessous pour qu’il l’indique.

```

4 D=0
5 G=0
6 for i in range (10):
7     P=randint(0,1)
8     if P==0:
9         D=D+1
10    else :
11        G=G+1
12 if .....:
13     print(".....")
14 else:
15     print(".....")

```

- Exécuter le programme plusieurs fois et noter le nombre de fois où il a réussi à rentrer à la base; en déduire la fréquence obtenue de réussite .

f =

Deuxième partie

Ecrire une fonction marchedurobot(n) qui simule un nombre de fois n la marche précédente et retourne la fréquence obtenue de réussite.

```

1 def marchedurobot (n):
2
3
4
5
6
7
8
9
10

```