

Modélisation et robotique

Exercices 1

Pour tous les programmes que vous écrierez, trouver un ou une camarade de classe qui accepte d'en simuler l'exécution pas à pas, c'est à dire en suivant chaque instruction tour à tour, à la lettre.

Pour simuler l'exécution des boucles, dans cette feuille d'exercice, il vous suffira de compter le nombre de répétitions.

1 Révisions du cours

Question A. Qu'affichera le REPL Python ?

```
a = 3
b = 2 * a
a = 4
print("valeur de b : %s" % b)
```

Question B. Dessiner le trajet du personnage. Sur un quadrillage sans obstacles, dessiner le trajet d'un personnage orienté au départ vers l'Est et se déplaçant selon le programme :

```
av()
av()
tg()
av()
av()
av()
td()
av()
```

Quelle est l'orientation finale du personnage ?

Question C. Même question, pour les quatre programmes suivants.

```
# Programme 1
for i in range(3):
    av()
    av()
    tg()
    av()
```

```
# Programme 2
n = 3
for i in range(n):
    av()
    av()
    tg()
av()
```

```
# Programme 3
tata = 1
toto = tata + 2
tata = 0
for i in range(toto):
    av()
    print(tata+toto)
    av()
tg()
av()
```

```
# Programme 4
x = 3
av()
for i in range(x):
    av()
    tg()
    av()
x = 10
```

2 Écriture de petits programmes

Question D. Chercher les pièces d'or! Écrire un programme pour que l'héroïne ramasse le tas d'or puis s'oriente vers l'Ouest.

Question E. Fonction demi-tour. Définir une fonction permettant à l'héroïne d'effectuer un demi-tour. L'utiliser pour que l'héroïne de la figure 1 (en position initiale) aille jusqu'à l'obstacle à l'Est et revienne à sa position initiale. Que faut-il faire pour qu'elle retrouve également son orientation initiale ?

Question F. Avancer de cinq cases. Définir une fonction `av5()` qui avance de cinq cases. Donner une solution sans boucle et une avec boucle.

Question G. Aller/retour. Définir une fonction `ar5()` qui avance de cinq cases et revient en arrière de cinq cases Utiliser des boucles.

Question H. Reculer. Définir une fonction `re5()` qui recule de cinq cases. Le personnage doit être dans la même orientation qu'au départ.

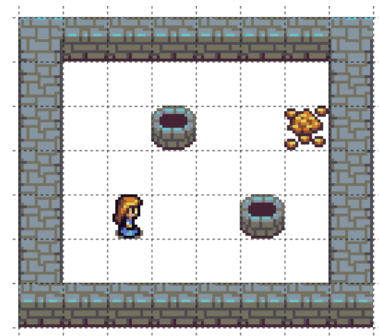


FIGURE 1 – Ramasser l'or

Question I. Cinq salades. Définir une fonction `plante5()` qui plante une ligne de cinq laitues. Bravo, ça pousse vite!

Question J. Avec paramètre (exercice complémentaire). Définir une fonction qui avance d'un nombre de cases donné par un paramètre.

3 Écriture avec des salades

Notre personnage, un peu facétieux, veut écrire dans son jardin avec des laitues, comme sur l'illustration (figure 2). Vous pouvez considérer qu'il se trouvait au départ sur une case avec une centaine de laitues et les a toutes ramassées.

Vous disposez d'un fonction `plante5()` et d'une fonction `plante2()` qui plantent respectivement une ligne de 5 et une ligne de 2 laitues.

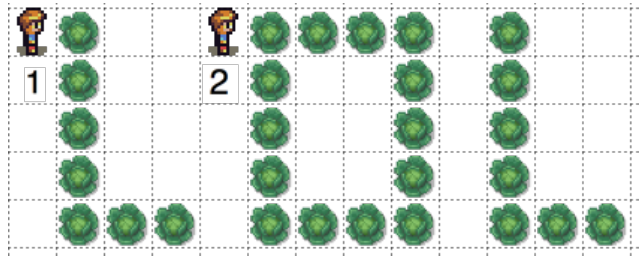


FIGURE 2 – Land art

Question K. Écrire un L. Le personnage est orienté vers l'Est en position 1 avec toutes ses laitues dans son panier. Écrire une fonction qui lui fait planter le L et l'amène à la position 2.

Question L. Écrire un O. Même question pour le O. Que faut-il faire pour terminer l'exercice?

4 Boucles imbriquées

Question M. Le jeu des différences. Quelle est la différence entre ces quatre programmes?

```
# Programme 1
for i in range(3):
    av5()
    av5()
    tg()
```

```
# Programme 2
for i in range(3):
    tg()
    for j in range(9):
        av()
    av()
av()
```

```
# Programme 3
for i in range(3):
    for j in range(5):
        for k in range(2):
            av()
    tg()
```

```
# Programme 4
av()
for i in range(3):
    av()
tg()
av()
```

Question N. Plantation de salades. Il reste plein de salades dans le panier de notre personnage. Servez-vous en pour remplir cet enclos en pierre!

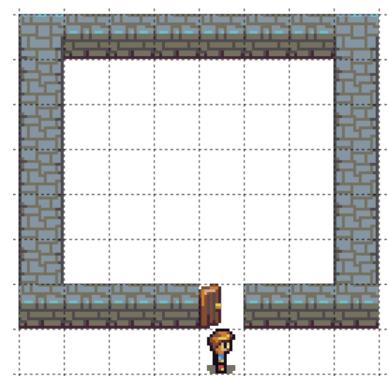


FIGURE 3 – Planter l'enclos