





PROGRAMMATION D'UN PETIT ROBOT: TI-ROVER



Nous allons dans cette séquence apprendre à programmer un petit robot grâce au Rover et au langage de programmation Python.



PREMIERS PAS

1. **Allumer** le Rover (interrupteur sur le dessus) et la calculatrice
2. **Cliquer sur** :  sur la calculatrice, puis sur :  sur : 
3. **Sélectionner** le programme AVANCE à l'aide des flèches, puis, après avoir posé le Rover par terre avec suffisamment d'espace autour, cliquer sur : 
4. **Noter** ce que le Rover a réalisé avec précision (vous disposez d'un mètre ruban pour mesurer les déplacements).



Le Rover a avancé en ligne droite de 1m50, puis il s'est arrêté

ANALYSE DE PROGRAMMES

Programme AVANCE:

Voici ci-contre le programme que vous venez d'exécuter.

Après avoir analysé les lignes de ce programme, proposer une modification afin que le Rover avance de 2 mètres.



Vérifier que cela fonctionne en modifiant le programme sur la calculatrice.



Programme AVANCE 2:

Voici ci-contre le programme que vous allez exécuter.

Après avoir analysé les lignes de ce programme, proposer une hypothèse sur ce que le Rover va faire:

Le Rover va avancer de 1 m puis tourner de 180° sur la droite, il va ensuite avancer de nouveau de 1m et faire de nouveau un demi tour sur la droite.

Vérifier votre hypothèse en lançant AVANCE 2 sur la calculatrice.



```
ÉDITEUR : AVANCE2
LIGNE DU SCRIPT 0001
from ti_system import *
import ti_rover as rv
rv.forward(10)
rv.right(180)
rv.forward(10)
rv.right(180)
```

Hypothèse Validée?

oui

non

Mémento:

Commande	Signification
rv.forward(10)	le Rover avance de 10 x 10 = 100cm soit 1m
rv.backward(15)	le Rover recule de 1m50
rv.left(45)	le Rover tourne de 45° à gauche par rapport à son axe de symétrie
rv.right(120)	le Rover tourne de 120° à droite par rapport à son axe de symétrie
rv.stay(10)	le Rover s'arrête 10 secondes

À partir du memento ci-dessus, **proposer** un programme afin que le Rover dessine un triangle équilatéral de 1 mètre de côté en faisant une pause de 5 secondes après chaque virage et qu'il revienne à sa position initiale :



```
ÉDITEUR : A1
LIGNE DU SCRIPT 0010
from ti_system import *
import ti_rover as rv
rv.forward(10)
rv.right(120)
rv.stay(5)
rv.forward(10)
rv.right(120)
rv.stay(5)
rv.forward(10)
rv.right(120)_
```

Appeler le professeur et lui montrer votre programme.

POUR ALLER PLUS LOIN

Proposer un programme afin que le Rover dessine un rectangle de 1 mètre de longueur et de 50cm de largeur en faisant une pause de 2 secondes après chaque longueur et qu'il revienne à sa position initiale :

```

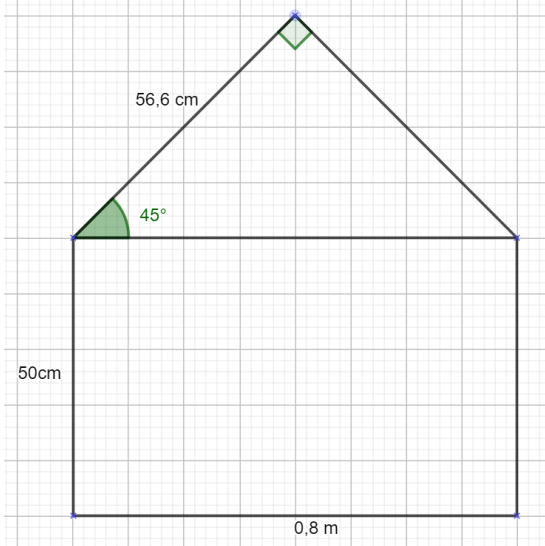
ÉDITEUR : RECTANGL
LIGNE DU SCRIPT 0001
_rom ti_system import *
import ti_rover as rv
i=1
while i<3:
  **rv.forward(10)
  **rv.right(90)
  **rv.stay(2)
  **rv.forward(5)
  **rv.right(90)
  **i=i+1
    
```

Proposer un programme afin que le Rover dessine un pentagone régulier de 50cm de côté sans faire de pause entre chaque virage et qu'il revienne à sa position initiale :

```

ÉDITEUR : PENTAGON
LIGNE DU SCRIPT 0001
_rom ti_system import *
import ti_rover as rv
i=1
while i<6:
  **rv.forward(5)
  **rv.right(72)
  **i=i+1
    
```

Proposer un programme afin que le Rover dessine une maison avec les dimensions du schéma ci-dessous sans jamais passer deux fois sur le même côté.



```

ÉDITEUR : MAISON
LIGNE DU SCRIPT 0001
from ti_system import *
import ti_rover as rv
rv.forward(5)
rv.left(90)
rv.forward(8)
rv.left(90)
rv.forward(5)
rv.left(90)
rv.forward(8)
rv.right(135)
rv.forward(5.66)
    
```

```

ÉDITEUR : MAISON
LIGNE DU SCRIPT 0022
rv.right(90)
rv.forward(5.66)
    
```

Proposer un programme afin que le Rover dessine un polygone régulier de telle sorte qu'on puisse choisir le nombre de face et la longueur d'une face.

```

ÉDITEUR : POLYG
LIGNE DU SCRIPT 0001
from ti_system import *
import ti_rover as rv
n=int(input("nbre de faces?"))
d=int(input("longueur d'1 coté?"))
i=0
rv.stay(5)
while i<n:
  **rv.forward(d)
  **rv.right(360/n)
  **i=i+1
    
```