

Ce programme simule le mouvement d'un robot simple. Un tel robot occupe une certaine position (x,y), il a 4 orientations {'Nord', 'Est', 'Sud', 'Ouest'}, il est initialisé à une position=(0,0) et une orientation='Nord', il peut tourner à droite et il peut avancer d'un pas . La classe Robot doit avoir les attributs de classe (Nom="Robot" et Version=0) et les attributs d'instances :

- Position=Point(x, y) représente la position du robot qui est en fait une instance de la classe point
- Orientation qui peut prendre comme valeur soit "Nord", "sud", "Est" ou "Ouest".

- 1) Définir une classe Point avec deux attributs x et y qui représentent les coordonnées d'un point.
- 2) Définir la classe Robot ayant les attributs de classe (Nom= "Robot" et Version=0) avec un constructeur qui initialisera les valeurs des attributs d'instances (position=Point(0,0) et orientation= 'Nord').
- 3) Enrichir la classe Robot avec les méthodes suivantes :
 - **TourneraDroite()** qui modifie l'orientation du robot sans changer sa position
 - **Avancer()** qui modifie la position du robot selon son orientation actuelle.
 - **AfficheRobot()** qui affiche la position et l'orientation du robot.

On veut améliorer ces robots en créant une nouvelle génération, RobotNG, qui hérite de la classe Robot. La classe RobotNG a les mêmes attributs que la classe Robot sauf les attributs de classe (Nom= "Robot nouvelle génération" et Version=1)

- 4) Définir la classe RobotNG.
- 5) Ajouter à la classe RobotNG les méthodes suivantes :

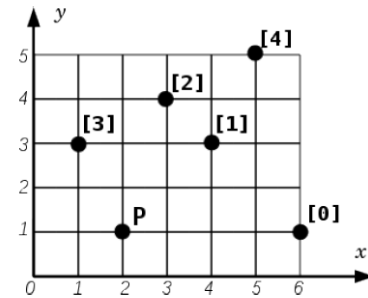
- **Avancer(npas)** : qui modifie la position du robot selon son orientation actuelle, le robot avance avec plusieurs pas (npas)
- **TourneraGauche()** qui modifie l'orientation du robot à gauche sans changer sa position
- **DemiTour()** : qui permet au robot de faire un demi-tour (180°)

Le RobotNG doit passer par un ensemble de cibles ou de points le plus proche de sa position. L'ensemble de cibles sera représenté par **une liste d'objets de la classe point** nommée **chemin** stockant les abscisses et les ordonnées des points qui composent le chemin suivi par le robot.

Exemple :

Chemin=

(6,1)	(4,3)	(3,4)	(1,3)	(5,5)
-------	-------	-------	-------	-------



Position actuelle du robot($X_P=2$, $Y_P=1$), $NextStep$ =point le plus proche c'est le point 3 de coordonnées (1,3).
Donc chemin suivi par le robot **point3 (1,3), point2 (3,4), point1 (4,3), point4 (5,5), point0 (6,1)**.

➔ On peut utiliser la distance entre deux points dans l'espace :

Distance(A,B)= $\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$ tel que A et B sont deux points de coordonnées (x_A, y_A) et (x_B, y_B).

6) Ajouter à la classe RobotNG une méthode **NextStep(Chemin)** qui permet de modifier la position du robot au point le plus proche parmi un ensemble de N points représentées le paramètre **chemin**

7) Ajouter à la classe RobotNG une fonction Trajet (Chemin) qui permet d'afficher le trajet suivi par le robot pour atteindre le point final (exemple points : point 3, point 2, point 1, point 4, point 0).