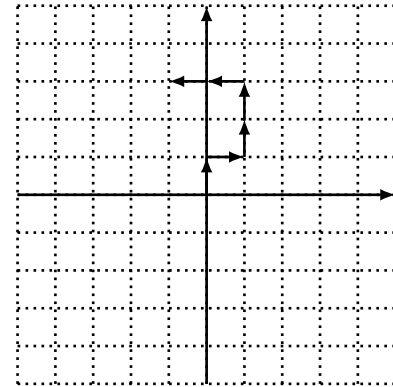


Une tortue se trouve au centre d'un carré de 10 pas de côté. Elle avance en faisant chaque pas au hasard dans une des quatre directions (gauche, droite, haut, bas). On se demande au bout de combien de pas, en moyenne, elle va sortir du carré.

On désigne par  $P$  le nombre de pas nécessaires pour sortir du carré. Ce n'est pas un nombre « fixe » : il dépend de l'expérience (comme le résultat du lancé d'un dé) ; c'est ce qu'on appelle une *variable aléatoire*<sup>1</sup>.

## 1 Étude préliminaire

On modélise l'expérience en plaçant la tortue à l'origine d'un repère orthonormé, chaque pas correspondant à un déplacement dans une des quatre directions. Un exemple est donné ci-contre.



1. Quelle est la plus petite valeur que peut prendre  $P$  ?
2. Montrer que  $P$  n'est pas borné, c'est-à-dire que  $P$  peut prendre des valeurs aussi grandes que l'on peut imaginer.
3. Quelle est la probabilité que la tortue sorte par l'arête nord du carré ?

## 2 Simulation d'une marche

1. Copier le fichier `marche.py` fourni par le professeur dans votre répertoire personnel, et l'exécuter. Que fait-il ?
2. Comment peut-on tester si la tortue est sortie du carré ? Modifier le programme pour qu'il s'arrête lorsque la tortue est sortie (aide : pour obtenir l'abscisse et l'ordonnée de la tortue, on pourra utiliser `xcor()` et `ycor()`).
3. Modifier le programme pour qu'il compte le nombre de pas, et qu'il affiche, à la fin, le nombre de pas effectués par la tortue avant de sortir du carré (aide : pour afficher une valeur (par exemple `n`) dans la console, on utilise `print(n)`).

## 3 Simulation de plusieurs marches

On désire maintenant simuler de nombreuses marches, pour pouvoir calculer la moyenne du nombre de pas nécessaires à la tortue pour sortir du carré.

1. Modifier le programme pour qu'il exécute 4 fois de suite la simulation (aide : on pourra utiliser la fonction `restart()`, qui permet de réinitialiser le programme).
2. Ajouter juste après les lignes `import` la ligne suivante. Elle permet de faire avancer la tortue plus vite, et de ne pas dessiner tous ses déplacements, ce qui accélère la simulation.  
`tracer(100, 0)`
3. Réaliser 100 simulations au lieu de 4.
4. Modifier le programme pour qu'il calcule la moyenne du nombre de pas faits par la tortue avant de sortir du carré.

---

1. Pas la peine de retenir ce mot : il n'est pas au programme.