



1. Les exercices du diaporama

1.1 DNB, Amérique du Nord 2019

```
In [ ]: # import de la bibliothèque gérant la tortue
        from turtle import *
        # un appui sur la touche tabulation après l'espace
        # suivant le import, affiche les fonctions disponibles !
        # de façon générale, un appui sur la touche <Tab> complète les mo

def carre():
    pendown()
    for _ in range(4): # quand le nom du compteur de boucle
        forward(cote) # est "muet" on écrit souvent _
        left(90)
    penup()

clearscreen()
setposition(...)
pensize(...)
cote = ...
for ...
    carre()
    ...
    ...
exitonclick()
```



1.2 La puce

Première version : on modélise le clavier par un axe d'origine la touche FA, chaque touche étant repérée par son abscisse (un entier relatif)

```
In [ ]: # import de la bibliothèque gérant les nombres aléatoires
        from random import uniform

        position = 0
        for _ in range(10):
            if uniform(0,1) < 0.5:
                position = position - 1 # en abrégé : position -= 1
            else:
                ...
        print(position)
```

Deuxième version : Réécrire le programme précédent sous forme de fonction et l'appeler 1000 fois dans une boucle, puis faire afficher la probabilité demandée.

```
In [ ]: # inutile d'importer de nouveau la bibliothèque random
        # c'est fait précédemment pour l'ensemble de la feuille.
        def dix_sauts():
            ...
            return position

        proba = 0
        for _ in range(...):
            if dix_sauts() == ...
                ...
        print("la proba d'être sur le FA initial est ", proba/...)
```



1.3 Les vecteurs colinéaires

Première version

```
In [ ]: # on fait simple, mais on profite quand même de
        # l'affectation en parallèle !
        xa, ya = 1, 1
        xb, yb = 2, 2
        xc, yc = 3, 3
        x_ab, y_ab = ...
        x_ac, y_ac = ...
        det = ...
        if ...
            print("les points sont alignés")
        else:
            ...
```

Deuxième version Ce programme fonctionne, pour changer les coordonnées des points, il suffit de changer les valeurs x_a , y_a , x_b , y_b , x_c et y_c dans le corps du programme.

Mais on préférera écrire des fonctions... Recopier le code du diaporama et le tester.

In []: