

# TP: Aléatoire en algorithmique

M1 MEEF maths

7 novembre 2022

Dans cette séance, on utilise le générateur pseudo-aléatoire fourni par Python pour expérimenter avec les algorithmes faisant intervenir le hasard.

```
from math import *  
from random import *
```

Quelques fonctions utiles pour produire des données aléatoires:

La fonction `random` donne un nombre à virgule flottante dans l'intervalle  $[0, 1[$  avec une distribution uniforme.

```
[random() for i in range(10)]
```

La fonction `randint` donne un entier aléatoire entre deux bornes incluses, avec une distribution aléatoire.

```
[randint(1, 6) for i in range(10)]
```

## Calcul d'aire

Première activité: programmer le calcul de  $\pi$  par la méthode de Monte-Carlo présentée dans le diaporama.

### Réalisation de l'expérience

La première étape est la réalisation de l'expérience avec un nombre de tirages donné. La fonction doit renvoyer la proportion de points arrivés dans le cercle.

```
def experience_cercle(nb_tirages):  
    ...
```

Une expérience avec différents nombres de tirages:

```
[experience_cercle(n) for n in [10, 100, 1000, 10000]]
```

### Calcul approché

La deuxième étape est d'en déduire une méthode de calcul approché de la valeur de  $\pi$ , en choisissant l'écart-type voulu.

```
def approx_pi(epsilon):  
    ...
```

Quelques tests:

```
approx_pi(0.1)
approx_pi(0.01)
approx_pi(0.001)
approx_pi(0.0001)
```

### Test de primalité de Fermat

Écrire le test de Fermat pour un nombre donné d'expériences:

```
def test_fermat(n, nb_exp):
    ...
```

Une liste de nombres premiers d'après ce test (exécuter plusieurs fois en faisant varier le nombre d'expériences):

```
[p for p in range(2,1000) if test_fermat(p, 2)]
```

Écrire un test de primalité classique:

```
def premier(n):
    ...
```

Pour comparer les deux, on peut énumérer des nombres pour lesquels le test probabiliste de coïncide pas avec le test déterministe. Là encore, on peut exécuter plusieurs fois le test pour observer ce qui se passe.

```
[p for p in range(2,1000) if premier(p) != test_fermat(p, 2)]
```

### Test de primalité de Miller-Rabin

Suivre les mêmes étapes que pour le test de Fermat puis comparer l'efficacité des deux méthodes.