

Semaine 8

Initiation à l'algorithmique et programmation

Revekka Kyriakoglou

Exercice : paradoxe des anniversaires

? A partir de combien de personnes dans une pièce des chances importantes qu'au moins deux aient la même date d'anniversaire ?

Exercice : paradoxe des anniversaires

? A partir de combien de personnes dans une pièce des chances importantes qu'au moins deux aient la même date d'anniversaire ?



Le paradoxe dit que dans un groupe de 23 personnes, il y a une chance sur deux qu'au moins deux personnes aient la même date d'anniversaire. Cette probabilité augmente avec le nombre de personnes présentes dans la pièce.

Exercice : paradoxe des anniversaires

• ? A partir de combien de personnes dans une pièce des chances importantes qu'au moins deux aient la même date d'anniversaire ?



Idée : On va simuler, en regardant plusieurs fois au bout de combien de personnes on a un doublon.

Exercice : paradoxe des anniversaires

? A partir de combien de personnes dans une pièce des chances importantes qu'au moins deux aient la même date d'anniversaire ?



Idée : On va simuler, en regardant plusieurs fois au bout de combien de personnes on a un doublon.

- Créez une liste `dates` dans laquelle vous pouvez ajouter des jours d'anniversaire choisis de manière aléatoire.
- Utiliser `randint` pour choisir une date :

```
from random import randint  
jour = randint(1,365)
```
- Lorsque vous trouvez un jour qui est déjà dans la liste `dates`, `break` votre boucle.

Solution

```
from random import randint

dates = []
compteur = 0
while True:
    compteur = compteur + 1
    jour = randint(1, 365)
    if jour in dates:
        break
    dates.append(jour)
print(compteur, 'personnes')
```

Exercise : Portée des variables

? Expliquer ce qui se passe dans ce programme. Qu'est-ce qui sera affiché dans le terminal ?

```
def f(x):  
    n = 1  
  
n = 3  
f(n)  
print('n_vaut', n)
```

Exercice : Portée des variables

? Expliquer ce qui se passe dans ce programme. Qu'est-ce qui sera affiché dans le terminal ?

```
def f(x):  
    n = 1  
  
n = 3  
f(n)  
print('n_vaut', n)
```



Le résultat est toujours n vaut 3 !

- 1 Variable Globale n** : définie à l'extérieur de toute fonction ($n = 3$) est une variable globale. Cela signifie qu'elle est accessible partout dans le programme, y compris à l'intérieur des fonctions, sauf si elle est masquée par une variable locale de même nom.
- 2 Variable Locale n dans la Fonction f** : À l'intérieur de la fonction f , nous avons une autre déclaration de n ($n = 1$) qui est une variable locale à la fonction f . Elle n'est accessible que dans le cadre de cette fonction et n'a aucune influence sur la variable globale n définie à l'extérieur de la fonction.
- 3 Passage de Paramètre x** : Lorsque vous appelez la fonction $f(n)$, la valeur de la variable globale n (qui est 3) est passée à la fonction f en tant que paramètre x . Notez que la fonction f ne modifie pas la valeur de x ; elle assigne simplement une nouvelle valeur à la variable locale n (qui devient 1).
- 4 Impression de la Variable Globale n** : Enfin, l'instruction `print('n vaut', n)` affiche la valeur de la variable globale n . Étant donné que la variable globale n n'a pas été modifiée par la fonction f , la sortie sera n vaut 3.