

1^{ère} NSI – Thème « #3 »

Programmation en Python

Pour aller plus loin...

Marc Biver, Septembre 2023

Lycée Fustel de Coulanges, Massy

*Les exercices qui suivent n'utilisent que les éléments de base de programmation Python que nous avons déjà vus en classe (seule exception : la notion de liste qu'on n'a pas encore couverte en cours) mais sont d'une difficulté supérieure à ce qu'on fait en séance. Je vous suggère de les résoudre dans l'environnement de votre choix (IDLE, Capytale...) et si vous le souhaitez de m'envoyer votre travail dans mon casier numérique (en n'oubliant pas de mettre votre nom et prénom dans le nom de fichier). **Important** : prenez bien soin de commenter votre code (notamment en spécifiant à quel exercice vous êtes en train de répondre) – je ne lirai pas de programme sans commentaire !*

1) Diviseurs d'un entier

L'utilisateur donne un entier supérieur à 1 et le programme affiche, s'il y en a, tous ses diviseurs propres sans répétition ainsi que leur nombre. S'il n'y en a pas, il indique qu'il est premier. Par exemple :

```
Entrez un entier strictement positif : 12
```

```
Diviseurs propres sans répétition de 12 : 2 3 4 6  
(soit 4 diviseurs propres)
```

```
Entrez un entier strictement positif : 13
```

```
Diviseurs propres sans répétition de 13 : aucun ! Il  
est premier
```

2) Approximation de e

Écrire un programme qui approxime par défaut la valeur de la constante mathématique e , pour n assez grand (*mais pas trop – attention au dépassement de capacité*), en utilisant la formule :

$$e \approx \sum_{k=0}^n \frac{1}{k!}$$

Pour cela, définissez la fonction factorielle et, dans votre programme principal, saisissez l'ordre n et affichez l'approximation correspondante de e .

3) Lancer de dés...

L'utilisateur donne un entier n entre 2 et 12, le programme donne le nombre de façons de faire n en lançant deux dés.

4) Lancer de dés... (*suite*)

Même problème que le précédent, mais avec n entre 3 et 18 et trois dés.

5) Lancer de dés... (*suite et fin*)

Généralisation des deux questions précédentes : l'utilisateur saisit deux entrées, d'une part le nombre de dés nbd (que l'on limitera à 10), et, d'autre

part, la somme s , comprise entre nbd et $6.nbd$. Le programme calcule et affiche le nombre de façons de faire s avec les nbd dés.

6) Générateur de mots de passe

Écrivez un programme Python qui génère un mot de passe aléatoire pour l'utilisateur. L'utilisateur devrait pouvoir spécifier la longueur du mot de passe. Le mot de passe généré devrait inclure des lettres majuscules, des lettres minuscules, des chiffres et des caractères spéciaux.

7) Palindrome

Écrivez un programme Python qui vérifie si un mot ou une phrase donnée par l'utilisateur est un palindrome. Un palindrome est un mot ou une phrase qui se lit de la même manière de gauche à droite et de droite à gauche, en ignorant la casse et les espaces.

8) Morpion

Implémentez le jeu du Morpion en Python, où deux joueurs peuvent jouer l'un contre l'autre. Le programme devrait afficher le plateau de jeu, permettre aux joueurs de placer leurs "X" ou "O", et vérifier s'il y a un gagnant ou un match nul.