Fonctions

Exercice 1. RACINES D'UN TRINÔME

- 1. Créer une fonction discriminant(a,b,c) qui prend en argument trois flottants a, b, c et qui renvoie le discriminant Δ du trinôme ax^2+bx+c .
- 2. Afficher le discriminant des polynômes suivants : $x^2 2x + 1$, $x^2 4$ et $x^2 + x + 1$.
- 3. Créer une fonction nb_racines(a,b,c) qui prend en argument trois flottants et qui renvoie le nombre de solutions réelles de l'équation $ax^2 + bx + c = 0$ dans le cas $a \neq 0$.

Par exemple, $nb_racines(1,-2,1)$ renvoie 1 nb_racines(1,0,-4) renvoie 2 $nb_racines(1,1,1)$ renvoie 0

4. Créer une fonction racines(a,b,c) qui prend en argument trois flottants et qui renvoie une chaîne de caractères avec les solutions réelles de l'équation $ax^2 + bx + c = 0$ dans le cas $a \neq 0$.

Par exemple.

racines(1,-2,1) renvoie « L'équation admet une unique solution : 1 » racines(1,0,-4) renvoie « L'équation admet deux solutions réelles : -2 et $2 \gg$

racines(1,1,1) renvoie « L'équation n'admet aucune solution réelle ».

Exercice 2. Vecteurs de l'espace

On se place dans un repère orthonormé direct $\mathcal{R} = (O, \overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k})$.

- 5. Créer une fonction produit_scalaire(u1,u2,u3,v1,v2,v3) qui prend en argument six flottants et qui renvoie le produit scalaire des vecteurs $\overrightarrow{u}(u_1, u_2, u_3)$ et $\overrightarrow{v}(v_1, v_2, v_3)$.
- 6. Créer une fonction norme(u1,u2,u3) qui prend en argument trois flottants et qui renvoie la norme du vecteur $\overrightarrow{u}(u_1, u_2, u_3)$.
- 7. Créer une fonction non_nul(u1,u2,u3) qui renvoie le booléen True lorsque le vecteur $\overrightarrow{u}(u_1, u_2, u_3)$ n'est pas nul, et False sinon.
- 8. Créer une fonction cos(u1,u2,u3,v1,v2,v3) qui prend en argument six flottants et qui renvoie le cosinus de l'angle $(\overrightarrow{u}, \overrightarrow{v})$.
- 9. Tester vos différentes fonctions en proposant quelques lignes de test pertinentes.

Rappel: Lorsque $\overrightarrow{u} \neq \overrightarrow{0}$ et $\overrightarrow{v} \neq \overrightarrow{0}$, on a

$$\overrightarrow{u} \cdot \overrightarrow{v} = ||\overrightarrow{u}|| \times ||\overrightarrow{v}|| \times \cos(\overrightarrow{u}, \overrightarrow{v}).$$

Exercice 3. MAXIMUM

10. Créer une fonction maxi(a,b,c) qui prend en argument trois flottants et qui renvoie le maximum de a, b et c.

Exercice 4. FACTORIELLE

11. Saisir et exécuter le programme suivant :

```
def carre(x) :
    resu=x**2
    return resu

for k in range(1,11) :
    print(carre(k))

for k in range(1,11) :
    print("Le carre de "+str(k)+" est "+str(carre(k)))
```

12. Créer une fonction fact(n) qui prend en argument un entier naturel n et qui renvoie le produit des n premiers entiers naturels non nuls :

$$1 \times 2 \times \cdots \times n$$
.

Par exemple,

fact(1) renvoie 1

fact(2) renvoie 2

fact(3) renvoie 6

fact(4) renvoie 24.

En mathématiques, on appelle ce produit « factorielle de $n \gg$ et on le note n! :

$$n! = 1 \times 2 \times \cdots \times n$$
.

Par convention, 0! = 1.

13. Écrire un script qui permet d'afficher le résultat suivant :

1! = 1

2! = 2

3! = 6

[]

20!=2432902008176640000

Exercice 5. Suite de Fibonacci

La suite (u_n) est définie par $u_0 = 0$, $u_1 = 1$ et $u_n = u_{n-1} + u_{n-2}$ pour tout $n \in \mathbb{N} \setminus \{0,1\}$.

u_0	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	 u_n
0	1	1	2	3	5	8	13	21	34	 $u_{n-1} + u_{n-2}$

- 14. Créer une fonction u(n) qui prend en argument un entier naturel n et qui renvoie la valeur u_n .
- 15. Afficher les 20 premiers termes de la suite (u_n) .