

Semaine n°11 du 09 décembre au 13 décembre

Informatique(Python) : cf exemples en annexe

- ⇒ Boucle `while` + Compteur.
- ⇒ Fonction récursive
- ⇒ Boucle `for` Calcul de somme.

Nombres complexes

- ⇒ Forme algébrique d'un nombre complexe, somme, produit, partie réelle et imaginaire, représentation géométrique.
- ⇒ Conjugué : définition, interprétation géométrique, propriétés, caractérisation des nombres réels et des nombres imaginaires purs avec le conjugué.
- ⇒ Module ($|z| = \sqrt{x^2 + y^2}$), expression avec le conjugué ($|z| = \sqrt{z\bar{z}}$), interprétation géométrique, propriétés
- ⇒ Inégalités triangulaires :

$$\forall (z, z') \in \mathbb{C}^2, \quad |z + z'| \leq |z| + |z'|$$

$$\forall (z, z') \in \mathbb{C}^2, \quad ||z| - |z'|| \leq |z - z'|$$
- ⇒ Notation $e^{i\theta}$, propriétés $e^{i\theta} \times e^{i\theta'} = e^{i(\theta+\theta')}$ et $\frac{e^{i\theta}}{e^{i\theta'}} = e^{i(\theta-\theta')}$, formule de Moivre, formules d'Euler.
- ⇒ Linéarisation d'expressions trigonométriques.
- ⇒ Argument d'un nombre complexe non nul, mise sous forme exponentielle d'un nombre complexe (technique de l'angle moitié), égalité de deux complexes sous forme exponentielle :

$$\forall (r, r', \theta, \theta') \in (\mathbb{R}_+^*)^2 \times \mathbb{R}^2, \quad r e^{i\theta} = r' e^{i\theta'} \iff \begin{cases} r = r' \\ \theta \equiv \theta' [2\pi] \end{cases}$$

- ⇒ Résolution des équations du second degré à coefficients réels, somme et produit des solutions. Résolution de l'équation $x^2 = a$ avec $a \in \mathbb{C}$

Primitives et intégrales

- ⇒ Primitives usuelles (cf formulaire) et reconnaissance des composées.
- ⇒ Intégrale d'une fonction continue sur un segment : définition, expression à l'aide d'une intégrale de l'unique primitive d'une fonction s'annulant en un point.
- ⇒ Propriété de l'intégrale : linéarité, positivité ([démonstration exigible](#)), croissance ([démonstration exigible](#)).
- ⇒ Intégration par parties ([démonstration exigible](#)). Application au calcul d'une primitive du logarithme ([démonstration exigible](#)).

- Calcul d'intégrales du type $\int_a^b P(x) \cos(\alpha x) dx$, $\int_a^b P(x) \sin(\alpha x) dx$, $\int_a^b P(x) e^{\alpha x} dx$ où P est un polynôme.
- Calcul d'intégrales du type $\int_a^b \cos(\alpha x) e^{\beta x} dx$ ou $\int_a^b \sin(\alpha x) e^{\beta x} dx$ à l'aide de deux intégrations par parties.

Remarques aux colleurs

- Merci aussi de poser une petite question d'informatique (cf Annexe).
- La notion de nombre complexe est nouvelle pour beaucoup de nos élèves.

Exemples de programmes informatiques

Exercice 1

Réaliser une fonction `DepasseValeur` prenant en paramètre un entier naturel M et renvoyant le plus petit entier naturel n tel que $2^n > M$.

```
def DepasseValeur(M):
    n=0 #initialisation du compteur
    while (2**n <=M):
        n=n+1 #incrementation du compteur
    return n
```

Exercice 2

Créer une fonction `somme` de paramètre n calculant $\sum_{k=0}^n k^3$

```
def somme(n):
    S=0
    for k in range(n):
        S = S + k**3
    return S
```

Exercice 3

Créer une fonction récursive `factorielle` qui prend en entrée un entier naturel n et renvoie le résultat de $n!$.

```
def factorielle(n):
    if n==0:
        return 1
    else :
        return n*factorielle(n-1)
```

Exercice 4

Créer une fonction récursive `suite` qui prend en entrée un entier naturel n et renvoie la valeur de u_n où (u_n) est la suite définie par :

$$\begin{cases} u_0 = \frac{1}{2} \\ \forall n \in \mathbb{N}, u_{n+1} = 3u_n + 2 \end{cases}$$

```
def suite(n):
    if n==0:
        return 1/2
    else :
        return 3*suite(n-1) +2
```