Programme de colle n^o 6 : Suites réelles, début des polynômes, révision.

Semaine du lundi 4 ***novembre****.*

Le programme de la semaine précédente est toujours au programme de cette semaine.

Suites remarquables (suite)

6.1 Suites arithmétiques : définition, raison, formule donnant le terme général. Somme de termes (consécutifs) d'une suite arithmétique.

6.2 Suite géométrique : définition, formule donnant le terme général d'une suite géométrique de raison $q \in \mathbb{R}$, somme de termes consécutifs d'une suite géométrique.

6.3 Suites arithmético-géométrique : définition, méthode pour déterminer le terme général d'une telle suite.

6.4 Suite récurrente linéaire d'ordre 2 à coefficients constants : définition. Soit $u = (u_n)_{n \geq n_0}$ une suite non identiquement nulle qui est géométrique de raison q, et soient a et b des réels. Alors u vérifie la relation de récurrence

$$\forall n \geq n_0, u_{n+2} = au_{n+1} + bu_n$$

si et seulement si q est solution de l'équation $x^2 - ax - b = 0$. Équation caractéristique associée à une relation de récurrence linéaire d'ordre 2 (à coefficients constants). Théorème donnant le terme général d'une suite vérifient une relation de récurrence linéaire d'ordre 2 dont l'équation caractéristique a un discriminant positif ou nul, exemples.

6.5 Démonstration du théorème sur les suites récurrentes linéaires d'ordre 2 (lemmes, puis une démonstration finale utilisant ceux-ci, voir cours).

Fonctions polynomiales

6.6 Notion de fonction polynômiale (ou polynôme). Ensemble $\mathbb{R}[X]$. Polynôme nul $0_{\mathbb{R}[X]}$, monôme, égalité des polynômes.

6.7 Soit $f(X) = \sum_{k=0}^{n} a_k X^k$ une fonction polynomiale. Alors, f est le polynôme nul si et seulement si :

$$\forall k \in [\![0, n]\!], a_k = 0.$$

6.8 Conséquence de la proposition précédente : théorème d'identification des coefficients, théorème d'unicité des coefficients.

6.9 Vocabulaire : coefficient de degré $k \in \mathbb{N}$ d'un polynôme, degré d'un polynôme, coefficient dominant, terme de plus haut degré, coefficient constant, termes d'un polynôme, polynômes unitaires. Racine d'un polynôme. Dérivabilité sur $\mathbb{R}$ des polynômes, la fonction dérivée d'un polynôme est un polynôme. Les polynômes sont indéfiniment dérivables sur $\mathbb{R}$.

Fonctions affines, fonctions polynomiales du second degré

6.10 Rappels sur les fonctions affines, vocabulaire, monotonie en fonction du coefficient directeur.

6.11 Fonctions polynomiales du second degré : forme développée, forme canonique. Soit $f(X) = aX^2 + bX + c$ un polynôme du second degré (a, b et c sont donc des réels avec $a \neq 0$). Si $b^2 - 4ac < 0$, alors f n'admet pas de racine. Discriminant d'une fonction polynomiale du second degré. Forme factorisée d'un polynôme du second degré de discriminant positif (puis, de discriminant quelconque), racines. Relations coefficients-racines. Variation et signe d'un polynôme du second degré.

Sera terminé à la rentrée : tracé de la courbes d'une fonctions polynomiales du second degré, en fonction de son discriminant et de son coefficient dominant.

Python

6.12 Fin des généralités : boucles while, affectation multiples. Calculs des termes d'une suite vérifiant une relation de récurrence d'ordre 2. Fonctions donnant la liste des termes d'une suite donnée par une relation de récurrence. Utilisation de boucles while pour détecter le premier rang d'une suite vérifiant une condition donnée.

6.13 Listes plus en détail : indices (positifs ou négatifs) et longueur d'une liste, modification d'une entrée d'un liste, ajout d'un élément en fin de liste, suppression d'un élément d'une liste. Fonction testant l'appartenance d'un élément à une liste. Fonction donnant le nombre d'occurrences d'un objet dans une liste, fonction donnant la liste des indices des occurrences d'un objet dans une liste.

Quelques questions de cours

Rappel : ces questions de cours seront au programme de la semaine suivante, tous les élèves sont donc concernés. Les premières questions de cours, plus rapides, devront être traitées aisément et accompagneront les suivantes.

1. (Rapide) Soit x un réel. Considérons l'implication (P) suivante : $(\exists a \in \mathbb{R}_+, x^2 > (x+1)^2) \implies x < 0$ (**toute** variante possible, au choix de l'interrogation). Donner la négation, la réciproque et la contraposée de (P).
2. (Rapide) Donner domaine de définition, de dérivabilité et tracer l'allure du graphe de la fonction usuelle suivante : (au choix de l'interrogation).
3. (Rapide) Toute définition des chapitres 1 à 5 (partie déjà traitée pour le 5) peut être posée en question de cours.
4. (Rapide) Toute proposition rapide à énoncer des chapitres 1 à 5 (partie déjà traitée pour le 5) peut être demandée, sans démonstration, en question de cours.
5. (Rapide) Soit $u = (u_n)_{n\in\mathbb{N}}$ une suite réelle. Démontrer l'équivalence : $(\forall n \in \mathbb{N}, u_{n+2} = 0) \iff (\forall n \geq 2, u_n = 0)$.
6. On pose $A = \{(x, y) \in \mathbb{R}^2 | 4x - y + 1 = 0\}$ et $B = \{(t + 1, 4t + 5) | t \in \mathbb{R}\}$. Démontrer que $A = B$ (toute variante assez similaire possible).
7. Énoncer et démontrer le résultat portant sur les sommes télescopiques.
8. Énoncer et démontrer le résultat portant sur les sommes géométriques.
9. Énoncer et démontrer l'inégalité triangulaire généralisée (prop. 56,(ii) du chapitre 3), utilisant le symbole $\sum$.
10. Soit $n \in \mathbb{N}$, calculer $\displaystyle\sum_{1\leq i,j\leq n} \min(i, j)$. *Élèves : Erreur typographique mineure (placement du "soit i...") dans l'annexe corrigée sur cahier de prépa.)*
11. Définir la notion de suite géométrique. Énoncer et démontrer la formule permettant de calculer la somme de termes consécutifs d'une suite géométrique (on traitera également le cas où la raison vaut 1).
12. Définir la notion de suite arithmético-géométrique. Calculer le terme général de la suite u définie par $u_0 = 2$ et $\forall n \in \mathbb{N}, u_{n+1} = 3u_n - 2$.
13. Soit $u = (u_n)_{n\geq n_0}$ une suite géométrique non identiquement nulle de raison q. Soit a et b des réels. À quelle condition la suite u vérifie-t-elle la relation de récurrence : $\forall n \in \mathbb{N}, u_{n+2} = au_{n+1} + bu_n$? Le démontrer.
14. Énoncer le théorème (47, chap. 4) permettant de calculer, dans certains cas, le terme général d'une suite récurrente linéaire d'ordre 2 (à coefficients constants). Calculer le terme général de la suite u donnée par $u_0 = 2$, $u_1 = 1$ et $\forall n \in \mathbb{N}, u_{n+2} = 4(u_{n+1} - u_n)$.
15. Énoncer le théorème (47, chap. 4) permettant de calculer, dans certains cas, le terme général d'une suite récurrente linéaire d'ordre 2 (à coefficients constants). Énoncer sans démonstration les lemmes permettant de démontrer ce théorème, et démontrer ce théorème dans le cas où le discriminant est strictement positif.
16. Énoncer la proposition (5, chap. 5) relative aux fonctions polynomiales identiquement nulles, puis le théorème d'identification des coefficients et le théorème d'unicité des coefficients. Démontrer la proposition 5 (méthode au choix).
17. Énoncer la proposition (5, chap. 5) relative aux fonctions polynomiales identiquement nulles, puis le théorème d'identification des coefficients et le théorème d'unicité des coefficients. Démontrer le théorème d'identification des coefficients.
18. Énoncer et démontrer la proposition et définition définissant le discriminant d'un polynôme du second degré. Énoncer la proposition et définition donnant la forme factorisée d'un polynôme du second degré de discriminant positif.
19. Énoncer la proposition et définition définissant le discriminant d'un polynôme du second degré. Énoncer et démontrer la proposition et définition donnant la forme factorisée d'un polynôme du second degré de discriminant positif.
20. Écrire le code d'une fonction Python d'entête `def U(n):` prenant en entrée un entier `n` et renvoyant en sortie le `n` ième terme de la suite u définie par $u_0 = 1$ et $\forall n \in \mathbb{N}, u_{n+1} = (n + 1)u_n + 2^n$.
21. Écrire le code d'une fonction Python prenant en entrée un entier naturel A et renvoyant en sortie le premier entier n pour lequel le n ième terme de la suite de Syracuse de paramètre A vaut 1. *On "admet" que ce programme devrait terminer :-).*
22. Écrire le code d'une fonction Python d'entête `def SuiteU(n):` prenant en entrée un entier `n` et renvoyant en sortie la liste `[` $u_0, \ldots, u_n$ `]` des $n + 1$ premiers termes de la suite u définie par $u_0 = u_1 = 1$ et $\forall n \in \mathbb{N}, u_{n+2} = 4u_{n+1} \ln(1 + u_n^2)$.
23. Écrire le code d'une fonction Python d'entête `def ListeOccurrences(L,a):` prenant en entrée une liste `L` et un objet python `a` et renvoyant en sortie la liste des indices des occurrences de `a` dans `L`.