

Programme de colle 20

Semaine du 16 mars 2026

Chapitre 16 : Limites de fonctions

Bien revoir les méthodes de calcul de limites, indispensables pour étudier la continuité.

Chapitre 17 : Probabilités conditionnelles et indépendance

➤ Cours à connaître

- ✓ Définition de probabilité conditionnelle et propriétés
- ✓ Formule des probabilités composées
- ✓ Système complet d'événements et formule des probabilités totales
- ✓ Formule de Bayes (classique et généralisée)
- ✓ Indépendance de deux événements et lien avec les probabilités conditionnelles
- ✓ Événements deux à deux indépendants et mutuellement indépendants.

➤ Exercices type

- ✓ Calculer une probabilité conditionnelle
- ✓ Déterminer une probabilité en utilisant les formules des probabilités composées, des probabilités totales et la formule de Bayes.
- ✓ Situations-type : tirage dans une (ou plusieurs) urnes, lancers de dés, tirage de cartes...
- ✓ Utiliser l'indépendance de deux événements.
- ✓ Utiliser l'indépendance d'événements.

Pensez à bien relire tout le chapitre 15, les points qui ne sont pas donnés explicitement dans ce programme peuvent être nécessaires pour déterminer certaines probabilités.

Chapitre 18 : Continuité

➤ Cours à connaître

- ✓ Définition continuité, continuité à droite et continuité à gauche en un point.
- ✓ Prolongement par continuité.
- ✓ Continuité et limites : composition avec une fonction continue et point fixe (il n'est pas attendu de connaître ces énoncés par cœur, mais de savoir les utiliser)
- ✓ Continuité sur un intervalle.
- ✓ Opérations et continuité.

➤ Exercices type

- ✓ Étudier la continuité d'une fonction en un point (calcul de limite)
- ✓ Étudier la continuité d'une fonction sur un intervalle (utiliser les fonctions usuelles)
- ✓ Étudier la possibilité d'un prolongement continu en un point.

Note aux colleurs. Le TVI n'est pas encore au programme pour cette semaine, il n'a pas encore été assez travaillé en cours

Python : simulation d'expériences aléatoires

Il faut connaître les commandes de base pour générer des nombres aléatoires :

- Importer la librairie Random : `import numpy.random as rd`
- Créer un nombre aléatoire de $[0; 1[$: `rd.random()`
- Créer un vecteur de n nombres aléatoires de $[0; 1[$: `rd.random(n)`
- Créer un entier aléatoire entre p et n : `rd.randint(p, n+1)` (attention : comme dans `range`, la dernière valeur n'est pas comptée). On peut aussi ajouter un paramètre pour avoir un vecteur avec plusieurs entiers, par exemple `rd.randint(1,4,10)` donne 10 entiers entre 1 et 3.

Il faut savoir les utiliser pour simuler une expérience aléatoire : lancer de dé, Pile ou Face, ou plusieurs expériences simultanées.

➤ Écrire une fonction `Piece` qui simule un lancer de pièce à Pile ou Face

```
import numpy.random as rd
def Piece() :
    alea = rd.random()
    if alea < 1/2 :
        return "Pile"
    else :
        return "Face"
```

Autre version :

```
import numpy.random as rd
def Piece() :
    alea = rd.randint(1,3) # entier aléatoire égale à 1 ou 2
    if alea == 1 :
        return "Pile"
    else :
        return "Face"
```

➤ Écrire une fonction `De` qui prend un entier n en paramètre et simule un lancer de n dés

```
import numpy.random as rd
def De(n) :
    alea = rd.randint(1,7,n)
    return alea
```