



Mathématiques discrètes

Projet : Génération des permutations

Consignes Le but du projet est de présenter une application dans laquelle les mathématiques discrètes jouent un rôle fondamental.

Le rendu final du projet consistera en un article destiné au grand public au format pdf de 800-1000 mots plus une annexe numérique, qui pourra contenir par exemple une démonstration interactive, une vidéo explicative et/ou des graphiques générés par code écrit par vous-même ; cette annexe sera rendue sous la forme d'un lien vers un dépôt en ligne. La forme exacte et la technologie utilisée pour l'annexe peut varier et est donc laissée au libre choix des étudiants. L'article et son annexe seront jugés non seulement sur le contenu mais aussi sur la clarté de la présentation, la qualité de rédaction, et la créativité.

Contenu Le sujet détaille quelques points à développer mais ceux-ci sont proposés comme point de départ de votre travail. Vous êtes encouragés à développer d'autres pistes en lien avec les mathématiques discrètes. De même, la bibliographie conseillée est un point de départ. Vous pouvez vous appuyer sur d'autres sources sur lesquelles vous porterez un œil critique et que vous prendrez soin de citer correctement.

Charte de bonne conduite Lisez attentivement la charte de bonne conduite. Portez une attention particulière à citer toutes vos sources, y compris les exemples et les images que vous utiliserez. L'équipe pédagogique sera très attentive à cet aspect lors de la correction.

Calendrier Consultez la page Moodle du cours pour les dates des principales étapes du projet.

Bref descriptif du sujet

Il existe plusieurs algorithmes pour énumérer toutes les permutations d'un ensemble d'éléments de taille n . Un algorithme est correct s'il énumère chaque permutation une et une seule fois.

Certains algorithmes sont basés sur les échanges successifs : C'est le cas par exemple de l'algorithme de Heap (1960) ou celui de Ives (1976). Certains algorithmes prennent une approche récursive, comme l'algorithme de Wells (1960), modifié par Boothroyd (1965). Ce ne sont que quelques exemples.

Bibliographie conseillée

- <http://www.princeton.edu/~rblee/ELE572Papers/p137-sedgewick.pdf>
- https://fr.wikipedia.org/wiki/Algorithme_de_Heap
- <http://villemin.gerard.free.fr/aNombre/MOTIF/PermutAl.htm>

Pistes de développement

1. Expliquer au moins deux algorithmes différents pour générer toutes les permutations.
2. Expliquer ce qu'est un permutoèdre.
3. Illustrer les algorithmes avec une animation Processing.