



DEVOIR MAISON II VERSION A

ECG2 MATHS APPLIQUÉES

1. Le but de ce devoir est d'écrire quelques programmes informatiques. Vous pouvez utiliser toutes les ressources qui vous semblent pertinentes¹, chercher de la documentation sur Internet ou dans des livres d'apprentissage de Python. En particulier, il pourra être utile de réviser la gestion des listes et la structure de graphe.
2. Vous pouvez travailler à plusieurs mais de préférence en petits groupes de 3 ou 4 étudiants. Faites alors attention à laisser chacun s'exprimer et participer à la rédaction du programme. Assurez-vous que chaque membre du groupe a compris la solution l'idée proposée avant de passer à la suite.
3. Dans ce cas, indiquez les autres étudiants avec qui vous avez travaillé.

Date: 8 Octobre 2024.

<http://louismerlin.fr>.

1. Chat GPT n'est pas une ressource pertinente.

CODAGE PAR DIFFÉRENCE.

Le codage par différence (delta encoding en anglais) permet de compresser un tableau de données en indiquant pour chaque donnée, sa différence avec la précédente (plutôt que la donnée elle-même). On se retrouve alors avec un tableau de données assez petites nécessitant moins de place en mémoire. Cette méthode se révèle efficace lorsque les valeurs consécutives sont proches.

Écrire une fonction Python d'en-tête `def delta_encoding` qui prend en paramètre un tableau non vide de nombres et renvoie

- La valeur initiale du tableau.
- Un tableau qui contient les différences des termes consécutifs du tableau passé en paramètre.

GESTION DES STOCKS

Un vendeur dispose de deux articles A et B . La taille de sa boutique ne lui permet que de stocker 40 articles de type A et 40 articles de type B . Tous les jours, 60 clients se présentent dans sa boutique et choisissent au hasard un article de type A ou de type B . On cherche à estimer la probabilité que le vendeur manque de stock.

1. Écrire une fonction Python qui simule les 60 achats des clients et qui renvoie 0 si tous les clients ont été satisfaits et 1 s'il existe au moins un client qui n'a pas pu obtenir l'article qu'il voulait.
2. À l'aide de la fonction précédente, donner une valeur approchée de la probabilité que le vendeur manque de stock.
3. Que faut-il en déduire ? Le vendeur doit-il chercher à agrandir sa boutique pour augmenter ses stocks ou peut-il au contraire se permettre de gérer ses stocks de cette façon-là ?

Remarque : À la fin du programme de cette année, nous aurons effectivement les outils théoriques pour calculer cette probabilité.

SOUVENIR DE VACANCES

Dans le cahier de vacances de cet été (exercice 55), nous avons étudié le mouvement d'une mouche qui se déplace dans votre appartement de deux pièces, selon le protocole suivant :

- Au tout début de l'étude, la mouche est dans le salon.
- Si la mouche est dans le salon, elle y reste avec probabilité $1/2$, sort de l'appartement par la fenêtre avec probabilité $1/4$ ou va faire un tour dans la chambre avec probabilité $1/4$.
- Si elle est dans la chambre, elle y reste avec probabilité $3/4$ et retourne dans le salon avec probabilité $1/4$.
- Une fois qu'elle est sortie, vous fermez la fenêtre et elle ne peut plus rentrer.

On cherche à comprendre la trajectoire de la mouche. Dans la suite de ce cours, nous aurons quelques outils théoriques pour s'y atteler. Ici, nous allons simplement utiliser Python et faire des simulations.

On dit que la mouche est dans l'état 1 si elle est dans le salon, dans l'état 2 si elle est dans la chambre et dans l'état 3 si elle est dehors.

1. Écrire une fonction Python d'en-tête `def simul_trajetoire(n)` : qui prend en argument un entier n et qui rend un tableau des n états consécutifs de la mouche. Ce tableau est donc constitué de n cases qui prennent les valeurs 1, 2 et 3. Noter que la première case de ce tableau est nécessairement 1 puisque la mouche démarre dans le salon.
2. Prenons $n = 1000$. Quelle est la dernière valeur (le dernier état de la mouche) dans le tableau retourné par la fonction `simul_trajetoire(1000)` ? Répéter l'expérience un certain nombre de fois. Pourquoi peut-on alors conjecturer que la mouche finira toujours par sortir de l'appartement ?
3. Proposer une méthode qui permette de calculer le temps moyen que la mouche passe dans l'appartement. Expliquer la construction du programme.

UNE ACTIVITÉ DU BDE (**PLUS DIFFICILE**)

Pour fêter l'arrivée des étudiants de 1^{ère} année, le BDE décide d'organiser une soirée "Killer". Le Killer est un jeu où chaque participant a pour cible un autre participant. L'organisateur choisit en secret des défis qui impliquent les participants et leurs cibles. Si l'un des participants réussit le défi, on dit qu'il "tue sa cible" et il récupère le défi que celle-ci aurait dû faire avec sa propre cible. Le vainqueur est le dernier joueur vivant. S'il y a n participants, l'organisateur doit donc fabriquer un cycle de défis de la forme

participant 1 \rightarrow participant 2 \rightarrow ... \rightarrow participant n \rightarrow participant 1.

Écrire une fonction Python qui prend en paramètre le nombre de participants et fabrique au hasard un cycle de défis.