1. *I2010 : langage C (3)*

# Tableau à 1 dimension

1. Ecrire un programme qui permet à l’utilisateur de calculer certaines statistiques sur les notes d’une classe. Le programme lit les notes au clavier via la fonction scanf (cf. eSyllabus chap. 6 ou la page de manuel man scanf). Il calcule la moyenne de la classe M(Xi), les écarts à la moyenne (Xi−M(Xi)), ainsi que sa variance, i.e. la moyenne des écarts au carré M( (Xi−M(Xi))² ). On supposera qu’il y a au maximum 50 étudiants dans une classe.

Essayez de minimiser le nombre de parcours de votre tableau de notes (deux parcours devraient suffire).

1. Un nombre premier est un nombre entier qui n’est divisible que par deux diviseurs : 1 et lui-même.

La recherche des nombres premiers a titillé la sagacité des mathématiciens depuis la nuit des temps. Au IIIème siècle avant J-C, en Grèce, le philosophe et mathématicien Eratosthène a imaginé une technique pour retrouver la liste des nombres premiers.

Son procédé, le crible d'Eratosthène, consiste, pour K allant de 2 à N, à parcourir la liste des nombres de K à N en supprimant tous les multiples de K. Les nombres qui n'auront pas été supprimés constituent la liste des nombres premiers inférieurs à N.

Il n’est bien entendu pas utile de (re)supprimer les multiples des nombres déjà supprimés. De même, les parcours peuvent s’interrompre dès que K atteint la valeur de la racine carrée de N.

Par exemple, si le programme doit rechercher les nombres premiers compris entre 2 et 10, les étapes seront les suivantes :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Etape 1 |  | **2** | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  |  |
| Etape 2 |  | **2** | **3** | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

La recherche peut s’arrêter là étant donné que 4\*4=16 est supérieur à 10. Les nombres premiers inférieurs à 10 sont donc : 2, 3, 5, 7.

Pour avoir une démonstration visuelle, rendez-vous sur le site :

[https://fr.wikipedia.org/wiki/Crible\_d%27%C3%89ratosth%C3%A8ne](https://fr.wikipedia.org/wiki/Crible_d'Ératosthène)

Il vous est demandé d’écrire un programme qui affiche les nombres premiers compris entre 2 et N grâce à la méthode décrite ci-dessus. Cette méthode ne nécessite ni division, ni multiplication !  
La valeur de N doit être strictement inférieure à 100 et être lue grâce à la fonction scanf. Veillez à vérifier que la donnée entrée par l’utilisateur est un entier valide.

# Tableau à deux dimensions

1. Ecrire un programme qui réserve une table de m lignes et n colonnes dont les éléments sont des caractères (m et n sont 2 chiffres lus sur *stdin* avec getchar). On remplit la table ligne par ligne avec des lettres majuscules, en commençant à la lettre 'A' et en continuant jusqu'à la lettre 'Z', puis en recommençant à 'A' et ainsi de suite jusqu'à ce que la table soit remplie.

*Exemple pour m = 5 et n = 7 :*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G |
| H | I | J | K | L | M | N |
| O | P | Q | R | S | T | U |
| V | W | X | Y | Z | A | B |
| C | D | E | F | G | H | I |

Si on remplit ensuite la table de la même manière mais colonne par colonne, on vous demande d'afficher les caractères, numéros de lignes et numéros de colonnes des cases ayant reçu la même valeur dans les deux remplissages.

*Avec l'exemple précédent, on obtient en sortie :*

A en ligne 0 et colonne 0

R en ligne 2 et colonne 3

I en ligne 4 et colonne 6

1. Modifier le programme précédent pour remplir la seconde table (ligne par ligne) avec des lettres choisies au hasard[[1]](#footnote-1). Le programme doit afficher les caractères, numéros de lignes et de colonnes des cases ayant reçu la même valeur que lors du remplissage précédent.
2. Pour terminer, votre programme doit afficher pour chaque ligne, la fréquence de chaque lettre. Par exemple, avec la table remplie au hasard.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| I | I | K | R | H | U | Y |
| J | K | H | L | H | V | R |
| I | Q | N | I | A | X | M |
| X | Y | M | G | B | Q | L |
| I | T | X | F | Y | W | A |

Le programme affichera :

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Ligne 1 : 0 0 0 0 0 0 0 1 2 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 1 0

Ligne 2 : 0 0 0 0 0 0 0 2 0 1 1 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0

Ligne 3 : 1 0 0 0 0 0 0 0 2 0 0 0 1 1 0 0 1 0 0 0 0 0 0 1 0 0

Ligne 4 : 0 1 0 0 0 0 1 0 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 1 1 0

Ligne 5 : 1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 1 0

# Bonus : rand()

Exécutez plusieurs fois votre second programme en entrant les mêmes données (nombre de lignes et nombre de colonnes) et comparez les résultats. Que remarquez-vous ?

Expliquez le comportement du programme et modifiez-le afin de lever la limitation observée.

1. . Pour obtenir un nombre aléatoire compris entre *min* et *max*, vous devez utiliser la formule générique :   
   min + (int)(rand()/(RAND\_MAX+1.0)\*(max-min+1)) où rand() et RAND\_MAX sont déclarés   
   dans le fichier *stdlib.h*. [↑](#footnote-ref-1)