

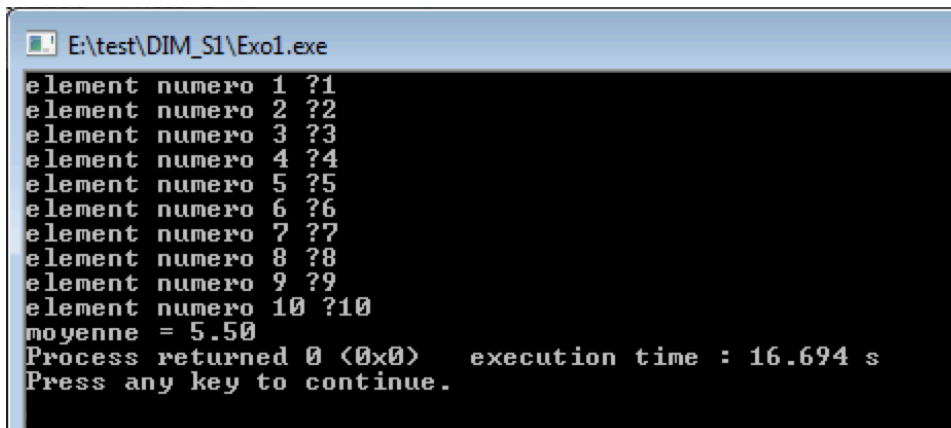
Algorithmique et langage C

Séance 1

Prérequis : Fondamentaux du langage C

Exercice 1 : Moyenne sélective d'un tableau.

Réaliser un algorithme en C permettant la moyenne des éléments positifs d'un tableau. On utilisera des entiers relatifs pour les éléments du tableau. Vous fixerez la taille du tableau à 10. Prévoir une saisie. Affichez le résultat avec deux chiffres significatifs. Vérifiez vos résultats à l'aide de jeux de tests.



```
E:\test\DIM_S1\Exo1.exe
element numero 1 ?1
element numero 2 ?2
element numero 3 ?3
element numero 4 ?4
element numero 5 ?5
element numero 6 ?6
element numero 7 ?7
element numero 8 ?8
element numero 9 ?9
element numero 10 ?10
moyenne = 5.50
Process returned 0 (0x0)   execution time : 16.694 s
Press any key to continue.
```

Que se passe-t-il si tous les éléments du tableau sont négatifs ?

Exercice 2 : Maximum d'un tableau

- Modifier l'algorithme précédent pour déterminer la valeur maximale d'un tableau.
- Modifier l'algorithme précédent pour déterminer la valeur maximale d'un tableau et la position de cette valeur dans le tableau (on supposera qu'il ne peut y avoir qu'une seule valeur maximale).

Exercice 3 : Renversement d'un tableau

Proposer un algorithme permettant de renverser les valeurs d'un tableau T de taille N connu (contrainte : ne pas utiliser de tableau intermédiaire). Réduire au maximum le nombre d'échanges.

```
E:\test\DIM_S1\Exo3.exe
element numero 1 ?1
element numero 2 ?2
element numero 3 ?5
element numero 4 ?4
element numero 5 ?8
element numero 6 ?9
element numero 7 ?10
element numero 8 ?11
element numero 9 ?14
element numero 10 ?18

Tableau initial :
1 2 5 4 8 9 10 11 14 18
Tableau final :
18 14 11 10 9 8 4 5 2 1
```

Exercice 4 : Remplissage aléatoire d'un tableau (jeu du démineur)

On dispose d'une fonction `alea(v)` qui fournit un nombre aléatoire entre 0 et $v-1$:

```
int alea(int v)

{return rand()%v;}

//mettre obligatoirement dans le main

srand(time(NULL));
```

Soit T un tableau de $N \times N$ caractères. Proposer un algorithme qui initialise le tableau T de manière aléatoire de sorte que K cases contiennent des 'X' et les autres cases des ' ' comme dans l'exemple suivant (tableau de 10×10) :

```
E:\test\DIM_S1\Exo4.exe
Combien de cases a initialiser (<=100) ?23
X      X      X
X      X      X X X
      X
X      X
X      X      X
X      X      X
      X X
      X X
      X X
      X X
```

Exercice 5 : Elimination des blancs d'une chaîne de caractères

Proposer un algorithme permettant d'éliminer les blancs d'une chaîne de caractères. On dispose d'une fonction `longueur(strlen)` qui retourne la longueur d'une chaîne (contrainte : ne pas utiliser de chaîne intermédiaire).

Exercice 6 : Tirage aléatoire dans un tableau

Proposer un algorithme effectuant le tirage aléatoire des valeurs d'un tableau. Toutes les valeurs doivent être tirées, et une valeur ne peut être tirée qu'une seule fois. On cherchera une solution efficace et rapide.

Exercice 7 : Analyse d'un texte

On suppose que l'on dispose d'un texte sous la forme d'un tableau de caractères, noté Texte, dont la taille est égale au nombre de caractères du texte.

- Proposer un algorithme permettant de calculer le nombre de mots du texte.
- Proposer un algorithme permettant de calculer le nombre d'occurrences d'un mot.

Exercice 9 : Calcul de la médiane des valeurs d'un tableau - Histogramme

- On dispose de 3 valeurs. Comment calculer la médiane de ces valeurs ? Comment ferait-on si on avait 100 valeurs ?
- On dispose d'un tableau de taille 100 dont les valeurs sont comprises entre 0 et 20. Proposer un algorithme permettant de calculer l'histogramme de ces 100 valeurs (l'histogramme est un tableau qui donne le nombre d'occurrences de chacune des valeurs du tableau).
- En utilisant cet histogramme, proposer une solution pour obtenir la médiane des valeurs du tableau.

Exercice 10 : Jeu de dé

On propose le jeu de dé suivant : chaque joueur lance un dé autant de fois qu'il le désire. S'il obtient le 1, il ne marque pas de point et passe la main à son adversaire. S'il décide d'arrêter (donc avant d'avoir obtenu un 1), il marque autant de points que de points obtenus par les différents lancers. La main passe ainsi successivement entre les deux joueurs, et le gagnant est le premier joueur à obtenir ou dépasser un score total de 100.

Proposer un algorithme de mise en œuvre de ce jeu avec les aménagements suivants :

- un joueur sera l'utilisateur du programme,
- l'autre joueur sera l'ordinateur,
- c'est l'utilisateur du programme qui commence à jouer,
- le tirage de dé sera simulé par un tirage aléatoire utilisant la fonction alea,
- la stratégie de l'utilisateur est laissée à son initiative. La poursuite éventuelle après chaque lancer du dé est décidée en fonction de la réponse à une question,
- la stratégie de l'ordinateur sera la suivante : il arrête de jouer s'il a obtenu 20 points (sans 1 donc) ou s'il a joué 5 coups successifs (sans 1 donc).