

Module : Algorithmique et Programmation

Contrôle continu (Durée : 2h)

Exercice 1 : (4 pt)

Le triangle de Pascal est obtenu en considérant que pour une ligne donnée :

- Le premier élément et le dernier élément sont égaux à 1.
- Les autres éléments sont obtenus au moyen de la relation : $T[l,c] = T[l-1,c] + T[l-1,c-1]$

Ecrire un algorithme permettant de générer et d'afficher le triangle de Pascal de taille n

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

Exercice 2 : (6 pt)

Ecrire un algorithme qui permet de saisir une matrice carrée M d'ordre m ($m \leq 10$) puis calcule et affiche le nombre d'éléments nuls, ensuite affiche la somme des éléments de la matrice, ceci en définissant :

1. Une procédure « SAISIE » qui permet de saisir les éléments d'une matrice.
2. Une fonction « NBR_NUL » qui permet de calculer le nombre d'éléments nuls dans la matrice
3. Une fonction « SOMME » qui permet de calculer la somme de tous les éléments de la matrice
4. Le programme principal.

Exemple :

Si la matrice saisie est :

L'algorithme affichera :

Le nombre d'éléments nuls est : 2

La somme des éléments de la matrice est : 37

1	0	3
5	7	6
0	12	3

Exercice 3 : (5 pt)

Ecrire un algorithme qui permet de saisir les éléments d'un tableau de caractères T de taille n, et d'inverser l'ordre de ses éléments et les afficher. (avec $n \leq 100$)

Exemple :

Supposant que le tableau d'origine est :

H	4	2	K	3	7
1	2	3	4	5	6

Après inversement, le tableau devient

7	3	K	2	4	H
1	2	3	4	5	6

Exercice 4 : (5 pt)

Ecrire un algorithme qui donne une approximation de e^x pour n donné, sachant que :

$$e^x \cong 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

Vous pouvez définir les fonctions qui vous seront utiles pour le calcul

Contrôle Terminal du Module : Algorithmique et Programmation

(Durée : 01h30)

Exercice 1 : (4 pt)

Soient les deux variables **ch** et **liste** définis comme suit :

ch = "programmation"

liste=[-3,-2,-1,1,2,3]

Quel est le résultat affiché par l'interpréteur en mode interactif

Les instructions sont indépendantes

Instructions	Résultats	Instructions	Résultats
ch.upper()		liste.insert(3,0) liste	
ch.isalpha()		liste[3:] liste	
ch.index('s')		liste[:3]='x','y','z' liste	
ch.count('M')		liste.reverse() liste	
ch.rfind("m")		liste[3:3]="M" liste	
Ch*2		liste.append(-4) liste	
ch[-4:]		liste[-4::-1] liste	
ch[1::4]		4 in liste	

Exercice 2 : (pt)

1. Ecrire une fonction « **REEMPLIR_LISTE** » qui prend comme paramètre un nombre n et qui **retourne** une liste **L** remplie de n entiers aléatoires compris entre 100 et 999 .
2. Ecrire une fonction « **SYMETRIQUE** » qui prend comme paramètre un entier m de 3 chiffres et qui retourne True si le nombre est symétrique et False dans le cas contraire
Exemple de nombres symétriques : 626, 929, 727,969
3. Ecrire le programme principal qui demande à l'utilisateur de définir la taille de la liste à créer, ensuite affiche uniquement les éléments de la liste qui sont symétriques

Exercice 3 : (6 pt)

1. Ecrire une fonction « **SAISIE** » qui permet de saisir les éléments d'une matrice carrée **M** d'ordre n et la retourner
2. Ecrire une fonction « **MAX** » qui retourne la valeur maximale des éléments d'une Matrice carrée **M** d'ordre n .
3. Ecrire une fonction « **SOM** » qui retourne la somme des valeurs de la **diagonale secondaire** d'une Matrice carrée **M** d'ordre n .
4. Ecrire un programme principal qui permet de tester les fonctions définies

Exercice 4 : (6 pt)

1. Ecrire une fonction nommée « **CODAGE_CESAR** » qui prend comme paramètre une chaîne de caractères **ch** et retourne une chaîne **sh** où chaque caractère de **ch** est décalé de trois crans dans l'alphabet. Par exemple a devient d, b devient e, c devient f.

On utilisera pour cela les méthodes `ord` et `chr` qui permettent la conversion des caractères en code ascii .

2. Ecrire un programme principal qui demande à l'utilisateur de saisir une chaîne de caractère, ensuite fait appel à la fonction `CODAGE_CESAR` qui permettra de coder la chaîne ensuite l'afficher. Le programme doit se réexécuter tant que l'utilisateur tape O

```
Entrer la chaine de caractères que vous voulez cryptez : ensa
la chaine de caractères après cryptage : hqvd
Pour crypter une autre chaine Tapez O pour sortir Tapez N : O
Entrer la chaine de caractères que vous voulez cryptez : algo
la chaine de caractères après cryptage : dojr
Pour crypter une autre chaine Tapez O pour sortir Tapez N : N
```