

Matière Langage C
[Contrôle Final – Janvier 2010 – (Durée : 1h30mn)]

N.B. Les réponses doivent être claires et précises. Les commentaires sont appréciés.

Exercice 1 (8 points)

Question 1 (4 points) :

Donner la suite des affichages produits par ce programme (Expliquer).

```
#include <stdio.h>
#include <stdlib.h>

int z ;

void affiche(int x , int y)
{
    switch( z )
    {
        case 1 : x = x++ + --y ; break ;
        case 2 : y = --x + y++ ;
    }
    printf("\nx = %d\ty = %d",x,y) ;
}

int transforme(int *px, int *py)
{
    return (*px>*py)?(*px)--:++*py ;
}

int main()
{int x = 1 , y = 2 , *px=&x, *py=&y ;

    z = 3 ;    affiche( x , y ) ;
    x *= y - transforme( &x , &y ) ;
    z = 2 ;    affiche( x , y ) ;
    y *= x - transforme( py , px ) ;
    z = 1 ;    affiche( x , y ) ;

    return 0;
}
```

Question 2 (4 points) :

Ecrire une fonction récursive qui affiche à l'écran la représentation en base 16 d'un entier long strictement positif X donné.

Exemples : Si **X = 20** alors la fonction affichera **14** ; Si **X = 30** alors la fonction affichera **1E**

Rappel : Le code ASCII du caractère 'A' est 65.

Exercice 2 (12 points)

Dans cet exercice, on considère les entiers de l'ensemble $\{-100,100\}$.

Une suite croissante d'un tableau T est définie par une suite d'éléments consécutifs de T trié par ordre croissant.

Exemple :

Si $T = (3 , 4 , -1 , 5 , 6 , 9 , 1 , 2)$ alors

- 1^{ère} suite croissante de longueur 2 est 3 , 4
- 2^{ème} suite croissante de longueur 4 est -1 , 5 , 6 , 9
- 3^{ème} suite croissante de longueur 2 est 1 , 2

La deuxième suite est la plus longue suite croissante de T.

- 1) Ecrire une fonction, nommée Sup_Suite_Croissante, qui retourne la longueur de la plus longue suite croissante d'un tableau T de N éléments donné.
- 2) Ecrire un programme C qui :
 - Remplit une matrice A carrée d'ordre M ($M \geq 5$) ligne par ligne par des entiers saisis au clavier. L'espace mémoire de A est alloué dynamiquement.
 - Puis construit un tableau T constitué par les maximums des lignes de la partie supérieure de la matrice A suivi par les minimums des colonnes de la partie inférieure de la matrice A. Les éléments de la diagonale ne seront pas traités. L'espace mémoire du tableau T est alloué dynamiquement.

Ex : $A = \begin{pmatrix} 10 & -1 & 3 & 0 & -12 \\ 13 & 0 & 1 & -2 & 4 \\ 6 & 37 & 19 & -21 & -1 \\ 11 & 15 & 1 & 1 & 5 \\ 34 & 9 & 11 & 2 & -1 \end{pmatrix} \rightarrow T = (3 , 4 , -1 , 5 , 6 , 9 , 1 , 2)$

- Enfin affiche la longueur de la suite croissante la plus longue de T (utiliser la fonction Sup_Suite_Croissante).

حظ سعيد

Matière Langage C
 [Contrôle de Rattrapage- Janvier 2010 - (Durée : 1h30mn)]

N.B. Les réponses doivent être claires et précises. Les commentaires sont appréciés.

Exercice 1 (4 points)

Copiez le tableau suivant et Complétez-le en indiquant pour chaque instruction si elle est correcte ou non. Dans le cas où c'est correcte, indiquez les valeurs des différentes variables modifiées. Dans le cas contraire justifiez avec une phrase sur la même ligne.

short A[3]={3,011,0xB}, *p1 ;

long a, b, *p2 ;

Instructions	Correct Oui/Non	Tableau A			p1	a	b	p2
		A[0]	A[1]	A[2]				
Indiquer les valeurs initiales des différentes variables								
a = *p1 , b = *p2 ;								
p1 = A , p2 = &a ;								
b = (*p1)++ ;								
*p2 = A[2] + (*p1)++ ;								
p1 = p2 ;								
p1 = &A[2] , p2=&b ;								
A[1] -= *p1+=*p2 ;								

Exercice 2 (6,5 points)

Une phrase est dite Palyndrome si tous les mots qui la constituent sont des Palyndromes. Un mot est dit Palyndrome s'il se lit à l'endroit comme à l'envers (SOS, RADAR et ELLE sont tous des mots Palyndromes). Un mot d'une phrase est délimité soit par un espace, soit par le début ou la fin de la chaîne.

Exemple : La phrase "SOS ELLE A 1 RADAR !!!" est palyndrome car les mots "SOS", "ELLE", "A", "1", "RADAR", "!!!" sont tous des mots palyndromes.

- 1) Ecrire une fonction, nommée **Mot_Palyndrome**, qui étant donnée un mot **CH** retourne 1 si **CH** est palyndrome ou 0 sinon.
- 2) Ecrire une fonction, nommée **Phrase_Palyndrome**, qui étant donnée une phrase **PH** retourne 1 si **PH** est palyndrome ou 0 sinon. Utiliser la fonction **Mot_Palyndrome**.

(Je rappelle que la fonction prédéfinie `strlen(PH)` permet de retourner le nombre des caractères de la chaîne PH sans compter le caractère de fin de chaîne '\0'. Elle est déclarée dans la bibliothèque <string.h>)

Exercice 3 (9,5 points)

Dans cet exercice, on manipule les entiers courts.

On appelle nombre de **Harshad**¹, un entier strictement positif **divisible** par la **somme de ses chiffres**.

Par exemple : 12 , 864 sont de tels nombres. En effet :

La somme des chiffres du nombre 12 est $2 + 1 = 3 \rightarrow 12 = 3 * 4$

La somme des chiffres du nombre 864 est $4 + 6 + 8 = 18 \rightarrow 864 = 18 * 48$

Les nombres de Harshad inférieurs à 1000 sont en nombre de **213**.

1. Ecrire une fonction récursive, nommée **Somme_Chiffres**, qui étant donné un entier N retourne la somme de ses chiffres.

2. Ecrire une fonction, nommée **Nombre_Harshad**, qui étant donné un entier N retourne 1 s'il s'agit d'un nombre de Harshad ou 0 sinon. Utiliser la fonction **Somme_Chiffres**.

3. Ecrire un programme C qui :

3.1 Remplit un tableau T des nombres de Harshad inférieurs à 1000. Utiliser la fonction **Nombre_Harshad**. L'espace mémoire pour le tableau T est alloué dynamiquement.

3.2 Puis affiche les éléments de T pairs (nombres de Harshad pairs) et leur nombre.

حظ سعيد

¹ : Le nom de Harshad leur a été donné par le mathématicien indien D. R. Kaprekar et signifie en sanscrit grande joie.