

# Corrigé Examen d'informatique

2005 - 2006

Durée : 2H

Tous documents autorisés

Les exercices sont indépendants. N'oubliez pas de numéroter vos copies et de rappeler sur chacune d'entre elles votre numéro d'identifiant.

## PROGRAMMATION EN C

### Exercice 1

Soit la séquence d'instructions suivante :

```
int f1(int n)
{
    printf("%d\n",n==0);
    return n;
}

int f2(int n)
{
    printf("%d\n",n=0);
    return n;
}

void f3(int *n)
{
    *n=*n+1;
}

int f4(int *n)
{
    *n=*n+1;
    return (*n);
}

int main()
{
    int a=1, b=2;
    a=f1(a);    b=f2(1);    printf("a=%d, b=%d\n",a,b);
    f3(&a);    b=f4(&a);    printf("a=%d, b=%d\n",a,b);
}
```

Solutions :

```
0
0
a=1, b=0
a=3, b=3
```

- dans f1, n affecté de la valeur de a vaut 1 est donc différent de 0, la comparaison n==0 vaut 0 et n n'est pas modifié,
- dans f2, n affecté de la valeur 1, mais l'affectation n=0 vaut 0 et modifie évidemment la valeur de n,
- a vaut 1 (valeur retournée par f1)
- b vaut 0 (valeur retournée par f2)
- dans f3, n est l'adresse de a, son contenu vaut 0 et est incrémenté de 1, donc après l'appel de f3, a vaut 2,
- dans f4, n est l'adresse de a, son contenu vaut 2 et est incrémenté de 1, qui est la valeur de retour, donc après l'appel de f4 a vaut 3 et b aussi.

**Exercice 2**

Écrire le code de la fonction `int strcmp(char *s1, char *s2)`; qui retourne 1 si les deux chaînes sont identiques et 0 sinon. Solutions :

```
#include<stdio.h>
#include<string.h>
int stringcmp(char * s1,char * s2)
{
    if(strlen(s1) !=strlen (s2))
        return 0;
    else
    while ( *s1 !='\0')
    {
        if(*s1 == *s2)
        {
            *s1++;
            *s2++;
        }
        else return 0;
    }
    return 1;
}
```

### Exercice 3

On donne le programme suivant. Il est chargé de calculer la moyenne des notes d'un étudiant. Compléter les instructions manquantes (indiquée par des .....) et compléter la fonction de calcul de la moyenne.

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

#define max 20

/* fonction de saisie des notes */
void Saisie(float * tab, int taille)
{
    int i;
        for(i=0;i<taille;i++)
        {
            printf("note n° %d ? ",i+1);
            scanf("%f", tab+i );
        }
}

/* fonction de calcul de la moyenne */
float moyenne(float * tab, int taille)
{
    int i;
    float somme,moy;
        .....
        .....
        .....
        .....
        .....
        .....
        .....

    return(moy);
}

/* programme principal */
int main( )
{
    float result ;
    float *note_etudiant;
    int Nb;

    do
    {
        printf("entrez le nombre de notes à traiter : ");
        scanf("%d", &Nb );
    }
```

```
}  
while ((Nb<=0)||(Nb>max));  
/* allocation dynamique du tableau de notes */  
  
note_etudiant =..... ;  
  
Saisie(note_etudiant,Nb);  
result = moyenne(..... ) ;  
printf("la moyenne de vos %d notes est %f\n", Nb, result);  
return 1;  
}
```

Solutions :

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

#define max 20

/* fonction de saisie des notes */
void Saisie(float * tab, int taille)
{
int i;
    for(i=0;i<taille;i++)
    {
        printf("note n° %d ? ",i+1);
        scanf("%f", tab+i );
    }
}

/* fonction de calcul de la moyenne */
float moyenne(float * tab, int taille)
{
int i;
float somme,moy;
    somme = 0;
    for(i=0;i<taille;i=i+1)
    {
        somme = somme + *tab;
        tab=tab+1;
    }
    moy = somme/taille;

return(moy);
}

/* programme principal */
int main( )
{
float result ;
float *note_etudiant;
int Nb;

do
{
printf("entrez le nombre de notes à traiter : ");
scanf("%d", &Nb );
}
while ((Nb<=0)|| (Nb>max));
/* allocation dynamique du tableau de notes */

note_etudiant =(float *)malloc(sizeof(float)*Nb);

Saisie(note_etudiant,Nb);
result = moyenne(note_etudiant,Nb) ;
printf("la moyenne de vos %d notes est %f\n", Nb, result);
return 1;
}
```

## Exercice 4

On dispose de la structure " date " suivante. Compléter les instructions manquantes (.....). Comment modifier la structure si on veut utiliser la date avec le nom du jour (lundi, mardi, ..., dimanche) en plus ?

```
#include <stdio.h>

#define Nb 3

typedef struct
{
    int jour;
    int mois;
    int annee;
} date ;

int main ( )
{
    date tab[Nb], *ptdate;
    int i=0;

    /* remplissage du tableau de dates */
    while ( i < Nb )
    {
        printf("\nEntrez une date (jj mm aaaa) : \n");
        scanf("%d %d %d",..... , ..... , .....);
        i++;
    }

    /* initialisation du pointeur sur la structure date */
    ptdate = tab;
    /* affichage du tableau de dates en le parcourant par un pointeur*/
    for(i=0; i<Nb; i++ )
    {
        printf("\nDate n° %d : %2d - %2d - %4d\n",i+1,..... , ..... , .....);
        ptdate++;
    }
    return 1;
}
```

Solutions :

```
#include <stdio.h>

#define Nb 3

typedef struct
{
    int jour;
    int mois;
    int annee;
} date ;

int main ( )
{
    date tab[Nb], *ptdate;
    int i=0;

    /* remplissage du tableau de dates */
    while ( i < Nb )
    {
        printf("\nEntrez une date (jj mm aaaa) : \n");
        scanf("%d %d %d",&tab[i].jour ,&tab[i].mois,&tab[i].annee);
        i++;
    }

    /* initialisation du pointeur sur la structure date */
    ptdate = tab;
    /* affichage du tableau de dates en le parcourant par un pointeur*/
    for(i=0; i<Nb; i++ )
    {
        printf("\nDate n° %d : %2d - %2d - %4d\n",i+1,(*ptdate).jour
        ,(*ptdate).mois,(*ptdate).annee);
        ptdate++;
    }
    return 1;
}
```