

Examen d'informatique

2005 - 2006

Durée : 2H

Notes de cours et de TD's uniquement autorisées

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 :

```
if (x>=y)
{
    printf("rouge");
}
else
{
    if(x<10)
    {
        printf("vert\n");
    }
    if(y>10)
    {
        printf("bleu");
    }
    else
    {
        printf("jaune");
    }
}
```

Déterminez l'affichage produit pour chacun des couples de nombres suivants.

- $x = 10$ et $y = 5$
- $x = 5$ et $y = 5$
- $x = 5$ et $y = 10$
- $x = 10$ et $y = 20$

Exercice 2

Soit la séquence d'instructions suivante :

```
1. S=0;
2. for(i=0;i<C1;i=i+1)
   {
3.   for(j=0;j<C2;j=j+1)
   {
4.     S=S+1;
   }
5.   printf("i=%d j=%d \n",i,j);
   }
```

1. Pour $C1=4$ et $C2=3$, quelle est la valeur de S à la fin de l'exécution ?
2. Pour $C1=4$ et $C2=3$, déterminer l'affichage produit.
3. En déduire pour n'importe quelles valeurs positives de $C1$ et $C2$ combien de fois sont exécutées les instructions 4 et 5 en fonction de $C1$ et de $C2$.

Exercice 3

Écrire le code de la fonction `char * strchr(char *s, char c)`; dont voici la page du manuel.

STRCHR(3) Manuel du programmeur Linux

STRCHR(3) NOM

`strchr` - Rechercher un caractère dans une chaîne.

SYNOPSIS

```
#include <string.h>
char *strchr (char *s, char c);
```

DESCRIPTION

La fonction `strchr()` renvoie un pointeur sur la première occurrence du caractère `c` dans la chaîne `s`.

VALEUR RENVOYÉE

La fonction `strchr()` renvoie un pointeur sur le caractère correspondant, ou `NULL` si le caractère n'a pas été trouvé.

Problème

On donne le programme suivant. Il est chargé de calculer les dépenses effectuées sur plusieurs journées. En cette fin de période de fête il est toujours bien de faire le bilan. Le programme est composé d'une fonction principale "main" et de plusieurs fonctions qui réalisent les actions suivantes :

- Saisir la date, le nombre et le montant des achats du jour.
- Calculer la somme et la moyenne des dépenses pour une journée ou pour un ensemble de journées.
- Afficher une date et la somme des dépenses pour cette date.

Complétez les instructions manquantes (indiqué par des ...), il y en 10 au total. Écrivez vos réponses directement sur la feuille d'énoncé. **N'oubliez pas de numérotez vos feuilles et de rappeler votre numéro d'identifiant.**

```
#include<stdio.h>
#include<stdlib.h>
/* structure definissant une date*/
typedef struct
{
    int jour;
    int mois;
    int annee;
}Date;
/* structure definissant les depenses d'un jour */
typedef struct
{
    Date date;
    int nb_dep; /* nombre de depenses du jour */
    float *depense; /* ensemble des depenses d'un jour
}Depense_jour;

/* Retourne un tableau de dépenses saisies par l'utilisateur, de taille donnée par
le parametre nb */
float * saisie_prix(int nb)
{
    float * tab;
    int i;
    /* Allocation dynamique du tableau*/

    ....
    /* Remplissage du tableau*/
    for(i=0;i<nb;i=i+1)
    {
        printf("prix %d\n",i+1);

        ....

    }
    return tab;
}
```

```
/* Retourne la somme des n premieres depenses contenues dans le tableau tab*/
float somme(float * tab,int n)
{
    int i;
    float S;
    S=0;
    for(i=0;i<taille;i=i+1)
    {
        S=S+tab[i];
    }
    return S;
}
/* retourne une structure Date dont les champs ont ete saisis par l'utilisateur */
Date saisie_date()
{
    Date d;

    ....
}

/* retourne une structure Depense_jour dont les champs ont ete saisis par l'utilisateur */
Depense_jour saisie_depense_jour()
{
    Depense_jour dj;
    int nb_kdo = 0;

    dj.date = ....

    while(nb_kdo<=0)
    {
        printf("Combien de cadeaux avez vous achetez ce jour là ?\n");
        scanf("%d",&nb_kdo);
    }
    dj.nb_dep= .....

    dj.depense = .....
    return dj;
}
```

```
/* affichage la date contenue dans la structure Date */
void affiche_date(Date d)
{
    printf("%d - %d - %d\n",d.jour,d.mois,d.annee);
}

/* affiche la somme totale des depenses d'un jour */
void affiche_total_jour(Depense_jour dj)
{
    ....
}

/* retourne la somme totale des depenses */
float somme_total(Depense_jour dj[100],int nb_j)
{
    int i;
    float total = 0;

    ....
    return total;
}

/* retourne la moyenne des achats par jour*/
float moy_total(Depense_jour dj[100],int nb_j)
{
    ....
}
}
```

```
/* Programme principal */
int main()
{
    /* tableau de structure, tab[i] est la structure modelisant les depenses du jour i*/
    Depense_jour tab[100];
    int n,i;
    /* Saisie du nombres de cases de tab */
    printf(" Sur combien de jours voulez vous gerer les depenses ? (au maximum 100)\n");
    scanf("%d",&n);
    /* Remplissage des depenses pour chaque jour */
    for(i = 0;i<n;i=i+1)
    {
        tab[i]=saisie_depense_jour();
    }
    /* affichage des sommes depensees par jour */
    for(i = 0;i<n;i=i+1)
    {
        printf("*****\n");
        affiche_total_jour(tab[i]);
    }
    /* affichage de la somme et de la moyenne des depenses */
    printf("\n Total sur %d jours = %f euros\n",n,somme_total(tab,n));
    printf(" Moyenne par jour = %f euros\n", .... );

    return 1;
}
```

UNIX

Tous les noms commençant par `fic` correspondent à des fichiers ordinaires de type texte.

Le résultat de la commande `ls` pour le répertoire courant est :

`fic0` (contient "uno, dos, tres")

`rep/`

Le résultat de la commande `ls` pour le répertoire `rep/` est :

`fic1` (contient "one, two,three")

`fic2` (contient "eins, zwei, drei")

`fic3` (contient "um, dois, très")

Remplissez le tableau suivant. Les instructions sont exécutées dans le répertoire courant et à la suite des autres.

Ajoutez votre numéro en haut de la feuille. Celle-ci devra être rendue avec le reste de vos copies.

Commandes	Erreurs signalées par le shell ?	Fichiers créés ou modifiés avec leur contenu	Résultat à l'écran
1) ls> fic1			
2) ls rep/ >> fic1			
3) fic > fic1			
4) wc -l fic1			
5) cp fic rep/			
6) mv rep dir			
7) cd rep			
8) ls dir/			
9) more dir/fic3			
10) cat fic1			