
Programmation en langage C

Contrôle du 25 janvier 2005. Durée 2 heures. Sans documents ni calculatrice.

Répondre sur l'énoncé.

Lisez attentivement tout le texte d'un exercice avant de commencer à le traiter, prenez votre temps avant de répondre.

Le correcteur tiendra compte de la clarté de la présentation (commentaires, *indentation* des programmes), de la justification des réponses, ainsi que de l'orthographe.

Rappelons que dans un programme toute définition de variable doit être suivie d'un commentaire expliquant son rôle, à moins que le nom de la variable ne soit à lui seul suffisamment explicite.

EXERCICE 1 - *Petits programmes simples*

1- Écrire un *programme* qui affiche les 10 premiers nombres multiples de 7.

2- Écrire une *fonction* `MinimumTableau` qui *retourne* la valeur minimum contenue dans un tableau `T0` de `N` entiers.

EXERCICE 2 - Questions de cours

1- Donner la déclaration d'une structure nommée `Point` représentant les coordonnées d'un point dans l'espace (3 dimensions) :

|

2- Donner la déclaration d'une variable `Tab` qui est un tableau de 50 structures `Point` :

|

3- Comment repère-t-on la fin d'un chaîne de caractères en langage C ?

|

4- Pourquoi l'extrait de programme ci-dessous est-il incorrect ?

```
int *T;
int x = *T + 1;
```

|

5-

```
#include <stdio.h>
#include <stdlib.h>
int main(void) {
    FILE *f = fopen("mama.txt", "rt");
    float sum = 0;
    while (!feof(f)) {
        char ligne[32];
        if (fgets(ligne,32,f) != NULL)
            sum = sum + atof(ligne);
    }
    printf("%f\n", sum);
    return 0;
}
```

Décrire ce que fait le programme ci-dessus.

EXERCICE 3 - *Utilisation des piles*

1- Rappeler les opérations sur les structures de données “pile” vues en cours, et décrire en une phrase chacune d’entre elles.

2- Soit le programme suivant :

```
#include <stdio.h>
#include "pile.h"
int main(void) {
    int T[] = { 1, 7, 3, 2, 8 };
    int N = 5; /* nb d'elements */
    int i;
    pile P;
    /* 1- Jongleries... */
    for (i=0; i < N; i++)
        empiler(&P, T[i]);
    for (i=0; i < N; i++)
        T[i] = depiler(&P);
    /* 2- Affichage du tableau */
    for (i=0; i < N; i++)
        printf("%d ", T[i]);
    printf("\n");
    return 0;
}
```

Décrire ce que fait ce programme et indiquer ce qu’il affiche.

3- Écrire un programme qui réalise exactement le même traitement que le programme présenté dans la question précédente, mais sans utiliser de pile, ni de pointeurs, ni de fonctions autre que main.