

**Examen Janvier 2018****EXAMEN**Matière: **LANGAGE DE PROGRAMMATION**

Enseignants: Ben Romdhan M, Challouf S, Ferjani N, Fkiri A, Haddaji H

Classes : TI.11, 12, 13, 14, 15, 16

Documents: **Non autorisés**Durée: **1h00'**

Date: /01/2018

*Il vous est demandé d'apporter un soin particulier à la présentation de votre copie.***EXERCICE 1 :****6 PTS**

Ecrire un programme C qui permet de lire à partir du clavier une suite de noms ne comportant pas d'espaces et d'afficher le nombre de noms ayant plus de dix caractères. La suite se termine lorsqu'on trappe le mot "FIN".

On supposera que chaque nom ne dépasse pas 20 caractères.

**NB** : Vous pouvez utiliser les fonctions de traitement des chaînes de caractères.

**Exemple** : Pour la suite de noms suivants :

Mohamed Ali  
Chokri  
Fatma  
Safia  
Salaheddine  
FIN

⇒ On a 2 noms qui ont plus de 10 caractères.

**EXERCICE 2 :****4 PTS**

Soit le programme C suivant :

```
#include <stdio.h>
void Aire(float A, float B, float C)
{
    C = A * B;
}
void main()
{
    float Large=10, Longueur=20, P=0;
    Aire(Large, Longueur, P);
    Printf(«Le périmètre du rectangle est %f » ,P) ;
}
```

1. Donnez les valeurs des variables Large, Longueur et P en sortie.
2. Quelles modifications doit-on apporter à la fonction et au programme principal pour qu'après l'appel de la fonction **Aire**, la variable P contient l'aire du rectangle.

**PROBLEME :****10 Pts**

On souhaite modéliser des opérations permettant de gérer les articles dans un stock de magasin à l'aide de tableaux à une dimension.

On considère pour cela un tableau *TabArt* de chaînes de caractères où les éléments représentent les codes des articles (10 caractères maximum), un tableau *TabStock* d'entiers représentant la quantité en stock de chaque article, et un tableau *TabPrix* de réels représentant les prix des articles.

Ainsi, un élément à l'indice *i* du tableau *TabStock* représente la quantité en stock de l'article de la case *i* du tableau *TabArt* et qui a comme prix la case *i* du tableau *TabPrix*.

On suppose que les tableaux ont une capacité maximale définie par une constante *max* donnée (*max* =100).

1. Ecrire une fonction *void LireArticles (char TabArt [max][10], int TabStock[], float TabPrix, int N)* permettant de remplir un tableau de chaînes de caractères *TabArt* de *N* articles, ainsi que les tableaux *TabStock* et *TabPrix* (pour chaque article lire son code, sa quantité en stock et son prix).
2. Ecrire une fonction *void AfficherArticles (char TabArt[max][10], int TabStock[], float TabPrix[], int N)* permettant d'afficher la liste des articles avec leurs quantités en stock et leurs prix.
3. Ecrire une fonction *int QuantitéStock(char TabArt[max][10], int TabStock[], char codeart[], int N)* permettant de rechercher la quantité en stock d'un article donné (*codeart*) et retourne sa quantité en stock s'il existe et -1 sinon.
4. Créer une procédure *void AjouterArticle (char TabArt[max][10], int Tabstock[], float TabPrix[], char codeart[], int qs, float px, int \*N)* permettant d'ajouter un article (*codeart*). Cette opération consiste à ajouter le code de l'article *codeart* à la fin du tableau *TabArt*, la quantité en stock *qs* dans le tableau *TabStock* et le prix *px* dans le tableau *TabPrix*.
5. Ecrire une fonction *void ApprovisionnerStock(char TabArt[max][10], int TabStock[], char codeart[], int qs, int N)* permettant d'approvisionner le stock d'un article en ajoutant une quantité donnée au stock existant. Cette fonction effectue une recherche de code de l'article dans le tableau *TabArt*, s'il existe alors elle ajoute la quantité *qs* au stock correspondant dans le tableau *TabStock*, s'il n'existe pas elle affiche un message d'erreur.
6. Ecrire une fonction *void MaxPrix(char TabArt[max][10], float TabPrix[], int N)* qui permet d'afficher le code et le prix de l'article le plus cher.
7. Ecrire une fonction *void SupprimerArticle (char TabArt[max][10], int TabStock[], float TabPrix[], char code[], int \*N)* permettant de supprimer un article stocké dans le tableau *TabArt* en fournissant code. Sa suppression entraîne la suppression de sa quantité en stock dans le tableau *TabStock* et de son prix du tableau *TabPrix*.
8. Ecrire un programme principale *main()* qui permet de :

- a) Remplir les tableaux TabArt, TabPrix et TabStock.
- b) Mettre à jour le stock d'un article donné par son code CD1 avec la quantité en stock *QTI* donnée.
- c) Ajouter un article dont le code CD2, la quantité QT2 et le prix PX sont données par l'utilisateur.
- d) Supprimer les articles dont la quantité en stock est égale à 0.
- e) Afficher tous les articles (code, prix et quantité).

***Bon Travail***