

Ex19 Révision structures et String

L'objectif de l'exercice est de réviser la gestion et l'intégration de chaîne de caractère dans une structure ou une union.

PREPARATION DU PROJET

L'exercice 2 sert de mode d'emploi pour la création du projet.

- Lancez le "Microsoft Visual Studio 2010
- Créez un nouveau projet :
 - Projets Visual C++
 - Application console Win32 (Dans la fenêtre modèle)
 - Sous emplacement: introduisez C:\LOGA_C\Nom
 - Décocher "Créer le répertoire pour la solution"
 - Sous Nom du projet : **Ex19**
- Paramètre de l'application :
 - Application Console
 - Projet Vide
- **Copie du fichier Ex19.c :**
Il faut copier le fichier Ex19.C de
K:\ES\Maitres-Eleves\SLO\Modules\SL124 LOGA\Exercices\Ex19
dans le répertoire Ex19. (Usage de l'explorateur Windows)
- Ajoutez au projet le fichier Ex19.c :
 - Depuis le Visual Studio 2010, sélectionnez "**Fichiers Sources**", avec un clic-droit obtenez le menu pour Ajouter un élément existant.
- Test de compilation : vous devez obtenir :

Génération : 1 a réussi, 0 a échoué, 0 a été ignore

REALISATION

Complétez le canevas fournis. Affichez le no de l'exercice, l'année et prénom, nom.

Introduisez :

```
#define _CRT_SECURE_NO_WARNINGS
```

Avant les #include.

TESTA

Soit un rectangle définit par sa hauteur et sa largeur. On souhaite dans un cas exprimer sa surface et dans l'autre son périmètre, ceci avec une seule structure. C'est la valeur du champ Text qui va déterminer le calcul à effectuer et le champ à affecter.

Le principe est le suivant :

| | |
|---------------------------|-----------------------------|
| char Text[16] = "Surface" | char Text[16] = "Perimetre" |
| double hauteur | double hauteur |
| double largeur | double largeur |
| double surface | double perimetre |

Le champ Text est unique dans la structure, les 3 autres champs forment 2 structures différentes dans une **union** contenue dans la structure globale du type S_DefRectangle.

1) DECLARATION DE LA STRUCTURE

Définissez en utilisant un typedef le type S_DefRectangle permettant de définir les deux systèmes de définitions des informations sur le rectangle.

2) INITIALISATION

Déclarer les variables DefRect1 et DefRect2 du type S_DefRectangle. Champ par champ, attribuez les valeurs suivantes :

Pour DefRect1 : "Surface" , 22.5, 37.2 surface par calcul

Pour DefRect2 : "Perimetre" , 19.4, 57.1 périmètre par calcul

3) UTILISATION

Réaliser une fonction **ShowInfo**, qui reçoit en paramètre un pointeur sur une structure du type S_DefRectangle.

Cette fonction doit afficher les informations sur le rectangle en donnant les noms des champs correspondants et leur valeur.

Dans le programme on appellera deux fois la fonction (Avec DefRect1 et DefRect2)

Aide : pour comparer les 2 chaînes vous aurez besoin de la fonction **strcmp** et pour initialiser de **strcpy**

Exemple de résultat.

```

D:\H\etCoursSW\SL124_LOGA\Exercices\ProjExVS2010\SolEx19\Debug\SolEx19.exe
Ex19 Christian HUBER
Test A ou B, Q pour Quitter
a
TestA:
Text = Surface Hauteur = 22.50 Largeur = 37.20 Surface = 837.00
Text = Perimetre Hauteur = 19.40 Largeur = 57.10 Perimetre = 153.00
Test A ou B, Q pour Quitter
  
```

TESTB

On souhaite transmettre octet par octet une trame définie par la structure suivante :

| | | | |
|------------------------|--------------------|--------------------------------|------------------------|
| STX(8 bits) signe { | MesLen (8 bits) | Message (Max 20 octets) | ETX(8 bits) signe } |
|------------------------|--------------------|--------------------------------|------------------------|

Le premier champ de la structure est STX (poids faible).

1) DECLARATION DE LA STRUCTURE ET DE L'UNION

Définissez en utilisant des typedef la **structure** du type **S_Frame** et l'**union** du type **U_Frame** permettant d'observer la structure octet par octet. Le champ permettant d'observer la structure octet par octet est un tableau d'**unsigned char** dont le nombre d'élément est **sizeof (S_Frame)**.

2) INITIALISATION

Déclarer les variables Frame1 et Frame2 du type **U_Frame**. Champ par champ (dans la partie active du programme), attribuez les valeurs suivantes :

Pour Frame1 : Message = "Hello !" MesLen = **strlen**(Message)

Pour Frame2 : Message = "Ecole Superieure" MesLen = **strlen**(Message)

Aide : vous aurez besoin pour l'initialisation champs par champs dans la partie active du programme de la fonction **strcpy** de string.h

3) UTILISATION

Réaliser une fonction **ShowFrame**, qui reçoit en paramètre un pointeur sur une **union** du type **U_Frame**.

Cette fonction doit afficher les informations de chacun des champs de la structure et aussi la valeur de chaque octet de la structure.

Les valeurs séparées sont affichée en fonction de leur nature. L'affichage octet par octet est réalisé en **hexadécimal**.

Dans le programme on appellera deux fois la fonction (Avec Frame1 et Frame2)

Remarque : avant l'appel de la fonction ShowFrame, il faut afficher la taille de la structure.

Exemple de résultat.

```

D:\H\etCoursSW\SL124_LOGA\Exercices\ProjExVS2010\SolEx19\Debug\SolEx19.exe
Ex19 Christian HUBER
Test A ou B, Q pour Quitter
B
TestB:
Taille S_Frame = 23
STX= { MesLen = 7 Message = hello ! ETX= }
7B 07 68 65 6C 6C 6F 20 21 00 CC CC CC CC CC CC CC CC CC CC CC 7D
STX= { MesLen = 16 Message = Ecole Superieure ETX= }
7B 10 45 63 6F 6C 65 20 53 75 70 65 72 69 65 75 72 65 00 CC CC CC 7D
Test A ou B, Q pour Quitter

```

Remarque : de par la nature de la structure, sa longueur correspond naturellement à la somme de la taille des champs. Le message n'occupant pas toute la place prévue on observe des valeurs CC.