

**Thèmes abordés :**

- La Pile 8086 : Utilisation des instructions PUSH, POP.
- La Pile 8086 : Manipulation de SP et BP.
- Le Data Segment : Etude de l'influence du contenu de « DS » sur l'accès aux données.
- Affichage en B8000h.
- Adressage.
- Procédures : Appel d'une procédure ; Passage de paramètres.
- Instructions logiques - Manipulation de Bits.
- Manipulation de fichiers.

**1 - Utilisation des instructions PUSH et POP**

Exercice 1 : Soient var1 et var2 deux variables de taille un octet. Ecrivez un code qui permet de permuter le contenu de ces deux variables en utilisant uniquement la pile.

Testez votre code avec les valeurs suivantes :

- var1 : 22
- var2 : 35

**2 – Manipulation de SP et BP – Procédures – Appel de procédure - Passage de paramètres**

Exercice 2 : Soient var1, var2, var3, var4 et Result cinq variables de taille deux octets. Ecrivez un code qui permet d'effectuer l'opération suivante :

$Result = 22 * (var1 + var2 + var3 + var4) + 35$

Pour ce code, on vous impose les contraintes suivantes :

- les 4 variables seront placées sur la pile (dans la procédure principale);
- une sous-procédure Somme permettra d'évaluer ladite expression ;
- un appel à la sous-procédure Somme sera effectué dans la procédure principale.

Testez votre code avec les valeurs suivantes :

- var1 : 75
- var2 : 45
- var3 : -33
- var4 : 95

Exercice 3 : Soient deux variables var1 et var2 de taille 2 octets. Ecrivez un code qui permet de permuter le contenu de ces deux variables. On vous impose d'utiliser une sous-procédure Swap qui effectuera la permutation. La sous-procédure Swap sera appelée dans la procédure principale.

Testez votre code avec les valeurs suivantes :

- var1 : 125
- var2 : 73

**3 – Affichage en B8000h**

Exercice 4 : Ecrivez un code qui permet d'afficher un message donné à l'écran. On vous impose de NE PAS UTILISER le service 09h INT 21h ni aucun service d'interruption d'affichage d'un caractère à l'écran. On écrira directement en l'adresse B8000h.

**4 – Manipulation des bits – Instructions logiques**

Exercice 5 : Soit var1 une variable de taille 2 octets et puissDeux une variable de taille un octet. On considère que le motif binaire stocké dans var1 est celui d'un nombre non signé. Ecrivez un code qui stocke dans puissDeux :

- 0 si var1 est impair
- 1 si var1 est multiple de 2
- 2 si var1 est multiple de 4
- 3 si var1 est multiple de 6
- .....
- 15 si var1 est multiple de 32768

Rédigez ce code sans utiliser l'instruction DIV et sans modifier le contenu initial de var1.

Testez votre code avec les valeurs suivantes :

- var1 : 25
- var1 : 7428
- var1 : 40960
- var1 : 23552

Exercice 6 : Soit var1 une variable contenant le motif binaire d'un entier non signé.

- Ecrivez un code qui permet de multiplier var1 par une puissance de 2 (2, 4, 8, 16, ...) sans utiliser l'instruction MUL.
- Ecrivez un code qui permet de diviser var1 par une puissance de 2 (2, 4, 8, 16, ...) sans utiliser l'instruction DIV.

Testez votre code avec les valeurs suivantes :

- var1 : 10101010b
- var1 : 11100101b

## 5 – Manipulation de fichiers

Exercice 7:

- Ecrire un code qui permet de créer un fichier texte et d'y écrire un message donné.
- Ecrire un code qui permet de lire le contenu d'un fichier texte et de l'afficher à l'écran.
- Ecrivez un code qui permet de supprimer un fichier donné.

Le nom du fichier est toujours fourni par l'utilisateur. La création ou la recherche du fichier a lieu dans le répertoire courant. Affichez un message d'erreur en cas de problème !

## 6 – Influence d'une modification de DS (INC DS) sur les données

Exercice 8 : Ecrire un code dans lequel on définit un tableau de 20 octets avec des valeurs de 1 à 20. Puis, on récupère l'élément 0 du tableau. Ensuite, modifiez DS (incrément de 1). Finalement, récupérez l'élément 0 du tableau.

## 7 – Exercice d'évaluation : Evaluation de la moyenne arithmétique des éléments d'un tableau et affichage du résultat à l'écran (via B8000h) et transfert du résultat dans un fichier.

Exercice 9 : Ecrivez un programme qui calcule la moyenne arithmétique des éléments d'un tableau d'entiers (signés, de taille un octet) donné (et hardcodé). Pour ce faire le programme sera pourvu d'un module « Somme » qui évalue la somme des « n » (fourni en argument) premiers éléments de ce tableau (également transmis en argument mais comment ?). Le programme affiche le résultat à l'écran, sans faire d'appel système (sans utiliser d'interruption d'affichage) ; puis le programme permettra d'écrire le résultat dans un fichier texte (de nom « moyenne.txt »).

NB : Le résultat sera considéré comme étant de type entier.