

03
15

15/12/2014

EXAMEN FINAL

EXERCICE N°1 (4 pts) (Toute réponse doit être justifiée)

1. Dites si l'adresse logique d'un opérande mémoire est toujours connue à la phase compilation.
2. Quelle est l'unité fonctionnelle du 8086 qui délimite la fin d'une instruction machine dans le 8086 ?
3. Deux instructions peuvent-elles s'exécuter en parallèle dans le 8086 ?
4. Dans quel type d'arithmétique les flags SF et OF sont-ils mis à jour ?
5. Parmi les instructions suivantes quelles sont celles qui modifient les flags de condition
 - > MOV AX, BX
 - > MOV AX, OFFSET Z ou LEA AX, Z
 - > INC Z
 - > INC AX
 - > CMP BX, 123D

EXERCICE N°2 (6pts)

Soit MATT1 une zone mémoire (tableau) de N*N octets, avec N un entier naturel. MATT1 contient les éléments d'une matrice donnée MATSOURCE, rangés en mémoire centrale ligne par ligne.

On veut construire MATT2 une zone mémoire (tableau) de N*N octets contenant les éléments de MATSOURCE rangés colonne par colonne.

Exemple :

$$N=3 \quad \text{MATSOURCE} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

MATT1 (1 2 3 4 5 6 7 8 9) et MATT2 (1 4 7 2 5 8 3 6 9)

1. Ecrire un programme assembleur qui réalise la construction de MATT2 à partir de MATT1.

15/12/2014.

EXERCICE N°3 (10 PTS)

Soit le programme suivant

Data segment

X DB 0

Tab1 db 16 DUP(1)

Tab2 db 255 DUP(0)

Tab3 db 64 DUP(2)

Y DW 20h

Data ends

CODE segment

✓ MOV CX, 0

✓ ADD CX, length Tab1

✓ ADD CX, length Tab2

✓ ADD CX, length Tab3

✓ MOV SI, 1

Boucle : MOV AL, [SI] *raiser*

CMP AL, 1

JZ suite

MOV [SI], 1 *raiser*

Suite : INC SI

DEC word ptr Y *raiser*

INC byte ptr X *raiser*

Loop Boucle

CODE ends

On supposera **DS=0100H**. Toutes les valeurs doivent être calculées en **HEXADECIMAL**

1. Calculer les attributs (adresse logique, adresse physique, longueur et taille) de chaque variable du segment DATA. (1,5pts)
2. Si le segment code est chargé à la suite du segment de données, calculer les adresses de début et de fin du data segment ainsi que l'adresse de chargement du segment code. En déduire la valeur du registre CS. (1,5pts)
3. Calculer le nombre total d'accès à la mémoire centrale suite à l'exécution de ce programme. (1pt)
4. Que fait ce programme ? (2pts)
5. Réécrire les instructions contenues dans la boucle loop afin de réduire le nombre d'accès à la mémoire centrale. (2pts)
6. Donner une image du segment DATA après exécution de ce programme. (2PTS)

Bon courage