

TD N3

(Programmation en langage d'assemblage du CPU 8086) (Manipulation des instructions de traitement de chaîne caractères)

Exercice 1

Ecrire un programme en langage d'assemblage qui calcule la somme des éléments d'une chaîne de mots de taille N se situant à l'adresse seg:adr. Le résultat est à mettre dans la case mémoire d'adresse seg:res

Exercice 2

Ecrire un programme en langage d'assemblage qui initialise une zone mémoire -débutant à l'adresse seg:adr- de N octets (tel que $N < 255$) avec les N premiers entiers consécutifs.

Exercice 3

Ecrire un programme en langage d'assemblage qui trie une chaîne de mots par ordre décroissant de ses éléments. La chaîne sera constitué de N éléments entiers se situant à l'adresse seg :adr.

Exercice 4

Ecrire un programme en langage d'assemblage permettant de remplacer dans une chaîne un octet x par un autre y. La chaîne est définie par son adresse de début seg :adr.

Devoir

- 1) Améliorer les exercices 2,4 pour qu'ils s'appliquent à des mots au lieu d'octets.
- 2) Ecrire un programme qui recopie le contenu d'une zone mémoire de taille N placée à l'adresse seg :src dans une zone mémoire de destination dont l'adresse de début est seg :dst, en inversant l'ordre des éléments.
- 3) Ecrire un programme qui recopie le contenu d'une chaîne source de taille N - dont l'adresse de début est seg :adrS - dans une chaîne destination située à seg :adrD. Les deux chaînes pouvant se chevaucher.

TD N4

(Programmation en langage d'assemblage du CPU 8086) (Sous-Programmes)

Exercice 1

Donner le programme en langage d'assemblage du CPU 8086 de la fonction C suivante :

```
int func( int v[], int w[] ) {  
    int i, s;  
    i = 0; s = 0;  
    while ((v[i] != 0) && (w[i] != 0)) {  
        s = s + v[i]*w[i]; ++i;  
    }  
    if (i >= 1) return s/i; else return 0;  
}
```

On considérera que les entiers et les tableaux sont représentés sur 1 octet et que la procédure appelée supprime les paramètres de la pile (convention pascal)

Exercice 2

Ecrire un sous-programme qui calcule le plus grand diviseur commun (pgdc) de deux nombres placés dans les registres AX et BX, et le met dans AX.

Ecrire un programme qui appelle cette procédure pour calculer le pgdc des nombres 16 et 24 et le range dans la case mémoire d'adresse seg :adr.

Exercice 3

Dans un premier temps, écrire un sous-programme RechMax qui recherche l'élément le plus grand dans une sous-chaîne ainsi que l'adresse de cet élément. Ce sous-programme utilisera les paramètres suivants :

- Registre SI (en entrée) : adresse de début du sous-chaîne
- Registre CX (en entrée) : adresse de fin du sous-chaîne
- Registre AX (en sortie) : valeur de l'élément le plus grand
- Registre BX (en sortie) : adresse de l'élément le plus grand

Ecrire un programme utilisant le sous-programme RechMax pour trier par ordre croissant une chaîne contenant des entiers codés sur 8 bits. Cette chaîne est définie par sa plage d'adresse : adrD-adrF.