

RATTRAPAGE

EXERCICE 1 :

La plupart des gens ne comptent sur leurs doigts que jusqu'à 10. En bon informaticien, vous pouvez faire mieux. Supposons que chaque bit représente un bit de valeur 1 si le doigt est levé et 0 si il est replié jusqu'à ce que les doigts commencent à se toucher. Si vous comptez avec les deux mains ? et si vous y ajoutez les pieds ? supposons maintenant que vous utilisez pieds et mains avec votre gros orteil gauche comme bit de signe ou complément à 2. quel est l'intervalle des nombres exprimables ?

EXERCICE 2 :

Pour trouver les racines d'un trinôme il faut calculer le discriminant  $\Delta$ . Selon que  $\Delta$  soit positif, négatif ou nul, on aura des racines réelles, des racines complexes ou une racine double.

Ou s'intéresse précisément a ce dernier cas :  $ax^2+bx+c=(x-\alpha)^2$ .

Ecrire un programme assembleur qui détermine si un trinôme a une racine double : dans ce cas il appelle la procédure de calcul de la racine sinon il va à la fin.

EXERCICE 3 :

A/ Soient deux machines M1 et M2.

M1 est une machine à 8 temps de séquençement et 2 cycles.

- Cycle Recherche et Indirection

- Cycle Exécution.

M2 est une machine à 6 temps de séquençement et 3 cycles :

- Cycle Recherche.

- Cycle Indirection.

- Cycle Exécution.

Quelle est la plus performante de ces deux machines ? Expliquez.

B/ Soit un ordinateur décrit en couches comme suit :

-chaque couche a des instructions « m » fois plus puissante que celles de la couche immédiatement inférieure, c'est à dire qu'une instruction de niveau « i » fait le même travail que « m » instructions de niveau « i-1 ».

si un programme de niveau 1 nécessite « k » secondes pour s'exécuter, combien de temps prendraient des niveaux équivalents de niveaux 2, 3 et 4.

C/ Les instructions d'un ordinateur ne possédant pas de compteur de programme ont le format suivant :

Code OP	Adresse de l'opérande	Adresse de l'instruction suivante
---------	-----------------------	-----------------------------------

Le code opération est sur 6 bits, et l'unité de mémoire a une capacité de 8192 mots.

1-Combien de bits doivent constituer un mot mémoire si une instruction est stockés dans un mot ?

2-Quelles sont les micro-opérations pour le Cycle Recherche ?

3-Donner les micro-opérations du Cycle Exécution de l'instruction JMP ?

D/ Qu'est-ce que l'adressage indexé ?

-Donner un exemple courant d'utilisation de ce type d'adressage

## SYNTHESE

### EXERCICE 1 :

Soient deux fichiers F1 et F2, Chacun de ces fichiers comporte un entier en représentation binaire. Chaque composant d'un fichier comporte un bit du nombre et le rang du bit dans le nombre. Les bits sont placés dans l'ordre croissant des rangs (C'est à dire du bit de poids le plus faible au bit de poids le plus fort).

Exp. : soit le nombre 1101,

Le fichier correspondant est : F 1 1 0 2 1 3 1 4 #

Etant données F1 et F2, il est demandé d'écrire le programme qui déterminer

- dans une liste G, le nombre binaire somme de ces deux nombres.
- et dans une liste M, le nombre binaire résultat de la multiplication par deux de la somme.

### EXERCICE 2 :

Soit F un fichier de caractères qui comporte des séquences de caractères terminées par deux caractères '\$' ('\$ '\$').

Chaque séquence comporte des sous séquences terminées par un seul '\$'.

Il est demandé d'écrire le programme qui donne :

- le nombre de séquences dans le fichier,
- Et pour chacune des sous séquences, le nombre de ces caractères.

Exp. :

F 'h' 'u' 'i' '\$' 'q' 'x' '\$' 'w' 'b' 'n' '\$' '\$' 'm' 'k' '\$' 'f' 'c' 'v' '\$' '\$' #

F comporte deux séquences.

La 1<sup>ère</sup> sous séquence comporte 3 caractères, etc.

### EXERCICE 3 :

A/ Ecrire le sous programme récursif SOMME qui calcule la somme de deux nombres entiers x et y. Ce programme doit utiliser correctement deux fonctions qu'on suppose prédéfinies :

- ADD1(z) qui ajoute 1 au nombre entier z.
- SUB1(z) qui soustrait 1 au nombre entier z.

B/ Ecrire le programme récursif qui définit le produit de 2 nombres entiers a et b quand on connaît les sous programmes ADD1, SUB1, et SOMME

**PREMIERE EPREUVE DE MOYENNE DUREE**

**Exercice 1**

Soit le programme suivant:

PROGRAM TEST

LIRE (C)

LIRE(B)

Y:=A+B+C+D/E

ECRIRE(Y)

END.

Ce programme s'exécute sur la machine de VON NEWMANN.

1. Calculez en octets la taille du code exécutable de ce programme. Justifiez votre réponse.
2. Donnez le cycle d'exécution de la quatrième instruction élémentaire.
3. Si ce programme devait s'exécuter sur le 8086 la taille de son code exécutable serait-elle la même que celle calculée en (1). Justifiez votre réponse.
4. A quoi se résume l'exécution d'une instruction de saut sur la machine de VON NEWMANN.

**Exercice 2**

X et Y sont 2 valeurs numériques entières s'écrivant sur n bits avec :

$X = \underbrace{1111 \dots 1111}_{n \text{ fois}}$  et  $Y = \underbrace{1111 \dots 1111}_{n \text{ fois}}$

1. Effectuez l'opération  $X + Y$ , donnez les valeurs des flags CF, OF, SF.
2. Sur combien de bits, en arithmétique non signé, s'écrira le résultat exact de l'opération effectuée en 1. Exprimez ce résultat en fonction de n.

**Exercice 3**

Soit une mémoire centrale de 1GO. (1GO =  $2^{30}$  O).

Cette mémoire est divisée en blocs selon la règle suivante:

Taille du bloc0 = Taille du bloc1 = 1MO.

Taille du bloci =  $2 * (\text{Taille du bloc}(i-1))$ ,  $i \geq 2$ .

1. Combien de blocs obtient on ?
2. Calculez en Hexadécimal les adresses de début et de fin du bloc 2 ainsi que celles du dernier bloc.
3. Si on note  $A_i$  l'adresse de début du bloci et  $B_i$  l'adresse de fin du bloci, établissez les formules de calcul de  $A_i$  et  $B_i$ .

**Exercice 4**

1. Pourquoi le bus du 8086 est-il multiplexé ?
2. Que se passerait-il si la file des instructions du 8086 était gérée selon la politique LIFO (dernier arrivé, premier servi).
3. Citez les relations qui existent entre : la taille de la mémoire centrale, la taille du bus de données, la taille du bus d'adresses, la taille des registres généraux.