

Nom : Prénom : Groupe :

Interrogation N°2

Exercice 1: (8 pts)

1. Répondez aux questions suivantes : (5 pts)

- A) Citez les étapes de fonctionnement d'une instruction dans le 8086 avec explication.
- B) Dans un tableau citez la différence entre l'architecture 32 et 64 bits ?

2. Pour chacune des lignes de la routine assembleur, identifiez les modes d'adressage utilisés (3 pts) :

- a. Immédiate
- b. Implicite
- c. Registre
- d. Adressage direct
- e. Adressage indirect
- f. Adressage indirect avec déplacement

Numéro de ligne	
1	strlen: mov esi, [esp+4]
2	mov eax, 0
3	@@: cmp byte ptr [esi+eax*1], 0
4	je @F
5	inc eax
6	jmp @B
7	@@: ret

Exercice 2 : (3 pts)

Ecrire la suite d'instructions pour réaliser les étapes suivantes :

- 1. copier le contenu de la case mémoire [1230h] dans CX
- 2. Comparer CX à 200
 - a. si # incrémenter CX et recommencer au point 2
 - b. si = copier CX dans AX et AX et effectuer un saut vers fin du programme

Exercice 3: Donnez l'adresse absolue de chacune des adresses suivantes : (4 pts)

- 3500 : AB00
- 0022 : FFFF
- 2000 :035C
- 4055 : 319E

Bon courage

Correction d'Interrogation N°1

Exercice 1 (8 pts) :

1. Les étapes de fonctionnement d'une instruction dans le 8086 avec explication : (2.5 pts)
 - La recherche dans la mémoire de l'instruction à exécuter adresse contenue dans le registre IP,
 - Charger l'instruction dans le registre d'instruction SI de l'UC et la décodée avec le décodeur,
 - Charger les opérands et l'opération depuis la mémoire dans le DS,
 - L'exécution de l'instruction par l'UAL et charger l'état de l'instruction dans le registre FLAGS et les opérands depuis le registre DS dans les registres généraux (AX, BX, CX, DX..) et le résultat dans le registre accumulateur AX,
 - L'adresse de l'instruction suivante chargée dans le registre IP
 - Stocker le résultat final dans la mémoire depuis l'accumulateur et le transfert des informations se fait à travers les bus de communications

2. La différence entre l'architecture 32 et 64 bits : (2.5 pts)

Architecture 32 bits	Architecture 64 bits
Utilise les mots mémoires codés sur 32 bits	Utilise les mots mémoires codés sur 64 bits
Adresser ou 2^{32} bits = 4 GB de mémoire	Adresser 2^{64} bits = 1 milliard GB de mémoire
Utilise les registres étendus de 8, 16 et 32 bits ; eax, ebx, ecx, edx, ebp, esp, edi, esi, eip, r8d,r15d plus les registres segment supplémentaires : FS, GS	Utilise les registres en mode réel 64 bits ; rax, rbx, rcx, rdx, rbp, rsp, rdi, rsi, rip, r8,r15 plus les registres étendus de 8, 16 et 32 bits
<i>Moins en terme de : Performance, Prise en charge de la mémoire, Compatibilité logicielle, Mises à jour futures</i>	<i>Plus en terme de : Performance, Prise en charge de la mémoire, Compatibilité logicielle, Mises à jour futures</i>
Utilise le mode d'adressage plat (flat mode memory) ainsi des nouvelles techniques comme la pagination de mémoire	Utilise le mode d'adressage plat (flat mode memory) ainsi des nouvelles techniques comme la pagination de mémoire

3. Pour chacune des lignes de la routine assembleur, identifiez les modes d'adressage utilisés: (3 pts)

- 1) Indirect avec déplacement
- 2) Immédiate
- 3) Indirect, Immédiate
- 4) Immédiate (JUMP vers registre FLAG)
- 5) Registre
- 6) Immédiate (JUMP vers registre FLAG)
- 7) Implicite

Exercice 2 (3 pts)

Ecrire la suite d'instructions pour réaliser les étapes suivantes :

1. copier le contenu de la case mémoire [1230h] dans CX
2. Comparer CX à 200
 - a. si # incrémenter CX et recommencer au point 2
 - b. si = copier CX dans AX et AX et effectuer un saut vers fin du programme

Mov CX, [1230h] ; copier le contenu de la case mémoire [1230] dans CX

Fct_comparer Proc near

Boucle

Cmp CX, 200 ; Comparer CX à 200

JNE Incrementer ; si # incrémenter CX et recommencer au point 2

JE Egal ; si égal = copier CX dans AX et effectuer un saut vers fin

LOOP Boucle

Incrementer :

INC CX ; incrémenter CX ; si CX est différent dans ce cas il sera incrémenter avec l'instruction INC et **il ne sera pas décrétementé comme compteur de la boucle dans ce cas**

Call fct_comparer

Egal :

Mov AX, CX ; Copier CX dans AX

JMP FIN

RET ; Retour au programme principal

endp

FIN : int 21h (hlt)

Ends

Exercice 3: L'adresse absolue de chacune des adresses suivantes : (4 pts)

- 3500 : AB00
- 0022 : FFFF
- 2000 :035C
- 4055 : 319E

Adresse physique =16x segment +déplacement

- ✓ **Convertir l'adresse de déplacement en décimal :** (AB00)₁₆= (43776)₁₀ , (FFFF)₁₆= (65535)₁₀
(035C)₁₆= (860)₁₀ , (319E)₁₆= (12702)₁₀
- ✓ **Les adresses physiques :**
 - 3500 : AB00 « Segment :déplacement » **Adresse physique**= 16x3500+43776=(99776)₁₀=(185C0)₁₆
 - 0022 : FFFF « Segment :déplacement » **Adresse physique**= 16x0022+65535=(65887)₁₀=(1015F)₁₆
 - 2000 :035C « Segment :déplacement » **Adresse physique**= 16x2000+860=(32860)₁₀=(805C)₁₆
 - 4055 : 319E « Segment :déplacement » **Adresse physique**= 16x4055+12702=(77582)₁₀=(12F0E)₁₆