



Architecture des ordinateurs

Deuxième année LMD

Examen du 28/01/2016

Durée : 1h30

**Exercice 01: 07 pts**

Soit le programme assembleur suivant :

Adresses	Instructions
100	MOV AX, 10
105	DEC AX
104h	MOV [200], AX
107	CMP AX, 0
10A	JE 103
10C	ADD AX, [200]
110	MOV AH, 4CH

3 1B8  
1 / 48  
3 A3  
3 13D  
2 / 7A  
3  
2

1. Donner les phases d'exécution pour chaque instruction, ainsi que le contenu des registres IP, RI, RTUAL et ACC. *Dr*
2. Donner le contenu de la MP après exécution.
3. Donner le mode d'adressage de chaque instruction.
4. Quelle est la taille de ce programme?

**Exercice 02 : 07 pts**

1. Ecrire un programme en assembleur 80x86 qui permet de concaténer deux chaînes de caractères déclarées dans le segment des données et d'afficher le résultat.
2. Ecrire un programme en assembleur 80x86 qui permet la concaténation et l'affichage par l'utilisation de deux procédures « Concaténer » et « Afficher ».

**Exercice 03 : 06 pts**

1. Quelle est la différence entre interruption DOS et BIOS, donner un exemple pour chaque cas.
2. Expliquer le rôle du contrôleur d'interruptions.
3. Comparer entre un traitant d'interruption et une procédure ou une fonction.
4. Expliquer le principe de fonctionnement d'une architecture pipeline, donner un exemple.



Quelques instructions du 80x86.

Symbole	Code op	Octets	Symbole
MOV AX, valeur	B8	3	AX ← valeur.
MOV AX, [adr]	A1	3	AX ← Contenu de l'adresse adr.
MOV [adr], AX	A3	3	Range AX à l'adresse adr.
ADD AX, valeur	05	3	AX ← AX + valeur
ADD AX, [adr]	03 06	4	AX ← AX + contenu de adr
SUB AX, valeur	2D	3	AX ← AX - valeur
SUB AX, [adr]	2B 06	4	AX ← AX - contenu de adr
SHR AX, 1	D1 E8	2	Décale AX à droite
SHL AX, 1	D1 E0	2	Décale AX à gauche
INC AX	40	1	AX ← AX + 1
DEC AX	48	1	AX ← AX - 1
CMP AX, valeur	3D	3	Compare AX et valeur
CMP AX, [adr]	3B 06	4	Compare AX et contenu de adr
JMP adr	EB	2	Saut inconditionnel (adr relatif).
JE adr	74	2	Saut si =
JNE adr	75	2	Saut si ≠
JG adr	7F	2	Saut si >
JLE adr	7E	2	Saut si ≤
JA adr			Saut si CF = 0
JB adr			Saut si CF = 1
MOV AH, 4C	B4 4C	2	