Ministère de l'Enseignement Supérieur et de la Recherche Scientifique Université Akli Mohand Oulhadj - Bouira -



Année: 2022/202

Département d'informatique

- J÷O30:X - X•XII: ∧I÷X:] 3|| X•V⊙•X

Module: Architecture des ordinateurs Examen: L2/S3 (1h+30m)

Note: ne pas utiliser les directives.

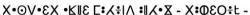
Exercice N°1(05 points):

Donner un programme assembleur 8086 qui demande d'entrer un entier *de deux chiffres décimaux* par le clavier, calcule la factorielle à l'aide d'une fonction *fact*, affiche le résultat sur l'écran. Le passage de paramètres et retour du résultat doivent se fait par la pile.

Solution:

```
JMP PROG
FACT:
POP BX : SAUVEGARDER IP
POP CX; PARAMETRE DANS CX
MOV AX.1
ICI:
  MUL CX
LOOP ICI
PUSH AX; RESULTAT DANS PILE
PUSH DX; CAS DE GRAND NB
PUSH BX; RECUPERER IP
RET
PROG:
MOV AH, 01
INT 21H
: PREMIER CHIFFRE DANS AL EXEMPLE: 5
SUB AL,30H; CODE ASCII DES CHIFFRES 30H--> 39H
MOV AH,00
MOV BL, 10
MUL BL
MOV BX, AX; PREMIER CHIFFRE*10 DANS BX, (50)
MOV AH. 01
INT 21H
; 2-IEME CHIFFRE DANS AL EXEMPLE 7
SUB AL,30H
MOV AH. 0
ADD AX,BX; LE NOMBRE DE 2 CHIFFRES DANS AX: 57
PUSH AX; passage par la pile
CALL FACT
POP CX: RECUPERER 16 BITS DE POIDS FORT DEPUIS LA PILE
POP BX : RECUPERER 16 BITS DE POIDS FAIBLE DEPUIS LA PILE
```

Université Akli Mohand Oulhadj - Bouira -





ونرامرة التعليــم العالي والبحث العلمي جامعة أكلي محند أوكحاج - البويرة -

```
; AFFICHAGE (VARIOUS METHODS ARE POSSIBLES)
ADD CL,30H
ADD CH,30H
ADD BL,30H
ADD BH,33H
MOV [100H], CX
MOV [102H], BX
MOV [104H], '$$'
MOV AH, 09
MOV DX, 100H
INT 21H
RET; FIN PROG
```

Exercice N°2(05 points):

Donner un programme assembleur 8086 pour déterminer le maximum des éléments d'un tableau stocké à partir de l'adresse mémoire 0xCC00: 100h. La taille du tableau est stocké dans la case mémoire 0xCC00: 200h. La taille de chaque élément est 16 bits. Sauvegarder le maximum dans la case 0xCC00: 300h.

Solution:

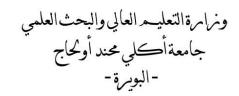
```
MOV AX, 0XCC00
MOV DS, AX
MOV CX, [200H]
SHL CX,1; chaque élément a deux cases
ADD CX, 100H
MOV SI, 100H
MOV AX, [SI]
DEBUT:
    INC SI
     INC SI
     CMP SI, CX
     JGE FIN
     CMP AX, [SI]
     JG DEBUT
    MOV AX, [SI]
JMP DEBUT
FIN: MOV [300H], AX
RET
```

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Akli Mohand Oulhadj - Bouira
X•••* *KIE E: *A÷IA : II *A•X - X•••

**OEO÷t -





Exercice N°3 (05 points):

Donner un programme assembleur 8086 ou *Intel x86 32 bits* pour diviser les éléments d'un tableau sur 2. La taille de chaque élément est 32 bits. Le tableau est stocké à partir de l'adresse 100h et la taille est 40 éléments. Ne pas utiliser l'instruction **div**.

MOV CX, 40
MOV ESI, 100H
ICI:

MOV EAX, [ESI]
SHR EAX,1; EAX <-- EAX/2
MOV [ESI], EAX
ADD ESI, 4; chaque élément a 4 cases
LOOP ICI

Exercice N°4 (05 points):

Soit les instructions suivantes. Déterminer les erreurs et proposer des corrections pour elles pour que les instructions soient conforme à l'assembleur 8086.

1	1- MOV [100H], [200H]	L'opération MOV ne peut pas se faire	MOV AL, [200H]
		entre deux cases mémoire, il faut que ça	MOV [100H], AL
		passe par un registre.	
2	2- MOV DS, 0XFF00	On ne peut pas faire des opérations	
		directement sur un registre segment, il	
		faut passer par un autre registre.	
3	MUL AX, BX	L'instruction MUL a un seul opérande.	MUL bx
4	PUSH [200H]	PUSH a besoin d'un opérande de 16	PUSH word [200h]
		bits.	
5	POP AL	POP a besoin d'un opérande de 16 bits.	POP AX
6	ADD AX, BX, CX	ADD a deux opérandes seulement	ADD AX, BX
			ADD AX, CX
7	OR AX, AL	OR s'effectue entre deux opérandes de	MOV BL,AL
		même tailles	MOV BH, 0
			OR AX, BX
8	LOOP AX	AX est un registre, ne pas l'utiliser	LOOP etq
		comme étiquette	
9	DIV DX:AX, BX	La division sur 16 bits, utilise	DIV BX
		automatiquement DX:AX et DIV a un	
		seul opérande	
10	INC [AX]	AX ne peut pas être utilisé pour	MOV DI, AX
		l'adressage	INC [DI]

جامعة البويرة

ونراسة التعليم العالي والبحث العلمي جامعة أكلي محند أوكحاج - البويرة -

- J+O30:X - X+X+: 3 3|X+ X3+VO+X

امتحان مقياس بنية الحواسيب

التمرين رقم 1 (05 نقاط):

هات برنامج مجمع 8086 الذي يطلب إدخال عدد صحيح مكون من رقمين عشريين بواسطة لوحة المفاتيح، ويحسب العاملي للعدد المدخل باستخدام دالة (برنامج جزئي) ، ويعرض النتيجة على الشاشة. يجب أن يتم تمرير المعطيات للدالة إرجاع النتيجة بواسطة المكدس.

التمرين رقم 2 (05 نقاط):

هات برنامج مجمع 8086 لتحديد أكبر عنصر لعناصر جدول المخزنة بداية من عنوان الذاكرة : CC00 الأكبر مخرن في خلية الذاكرة 200h: CC00. حجم كل عنصر 16 بت. احفظ العنصر الأكبر بعد تحديده في العنوان CC00: 300 h.

التمرين رقم 3 (05) نقاط):

أعطِ برنامج تجميع 8086 أو 808 Intel x86 بت لتقسيم عناصر جدول على 2. حجم كل عنصر هو 32 بت. يتم تخزين الجدول بداية من العنوان h100 والحجم 40 عنصرًا. لا تستخدم أمر div.

التمرين رقم 4 (05 نقاط):

ضع في اعتبارك التعليمات التالية المحاكية لمجمع 8086. يطلب منك تحديد الأخطاء واقتراح تصحيحات لها.

- 1- MOV [100H], [200H]
- 2- **MOV DS, 0XFF00**
- 3- MUL AX, BX
- 4- PUSH [200H]
- 5- POP AL
- 6- ADD AX, BX, CX
- 7- OR AX, AL
- 8- LOOP AX
- 9- DIV DX:AX, BX

10- INC [AX]