0370

National University of Computer and Emerging Sciences
Islamabad Campus

Computer Organization and Assembly Language (EE2003)

Sessional-I Exam

Total Time (Hrs):	1
Total Marks:	60
Total Questions:	5

Course Instructor(s):
Mr. Farrukh Bashir, Mr. Shams Farooq

Sections:
A, B, C ,D ,E ,F ,G

Date: Sep 24, 2024

Roll no

Do not write below this line.

Attempt all the questions.

INSTRUCTIONS

- Your final answer should be written with permanent pen, answer will not be acceptable with pencil
- Name all the identifiers in memory
- Consider all the memory starts at address **0x00000100H**
- You are supposed to submit your question paper back. NO Answer Book required

	Q-1	Q-2	Q-3	Q-4	Q-5	Total
Marks	3	7	2	10	5	25
Total	10	10	10	10	20	60

Question 1 [10 Marks]

Consider the following data declaration and fill in the given memory in hexadecimal (h).
Note: ASCII for 'A' = 041H, '1' = 031H

```
.data
    b1 byte 10011111b, 16d, 77q, '12AB', 012H,0ABH
    b2 sbyte "A"+1001b*10q, -1,0FFH, 255
    w1 dw 0ABCDH, 'AB','CD'
    d1 dword 'ABC',0ABCH
    q1 dq 123456ABCDH,'1234','ABCD'
```

9 f 8 4 2 1

0010000
10

64 32 16 8 4 2 1
0 1 0 0 1 1 0 1

4D 12 AB

41 041
11 90 98 90
00 D1 131 65
01 155

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0100	9f	10	4D	C	A	B	C	AB	D1	01	FF	CD	AB	B	A	D
0110	C	C	B	A	00	CB	0A	00	00	CD	AB	56	34	12	00	00
0120	00	9 00	3 00	29 00	10	00	00	00	00	D 00	00 C	B	A	B 00	B 00	00
0130	00	[illegible]	[illegible]													

16 | 255 | 1
16 | 17 | 1
1

111,1111

Nationa

National University of Computer and Emerging Sciences
Islamabad Campus

Question 2 [5+5 =10 Marks]

Update the given **memory** and **registers** after executing the following code.

Note: *Consider starting address at **0x00000100H** fill memory and registers in **HEX***

```
.data
     w1  dw 5H,6H,
             0ABH,8H,9H
      word 3H
      bptr LABEL BYTE
      wptr LABEL WORD
      dptr LABEL dword
      qary dq 012345678H
      count=2
      bary1 db 2 dup(count dup(1,2)), 0ABH
      count=3
      bary2 byte 12H, 2 dup(count dup(3))
```

A 10
B 11
C 12

//2

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0100	05	00	06	00	AB	00	08	00	09	00	03	00	78	56	34	12
0110	00	00	00	00	01	02	01	02	01	02	01	02	AB	C	03	03
0120	03	03	03	03												
0130																

```
mov eax,0
mov al,count
mov al, bptr
mov bx, [wptr]
mov ecx, dptr
mov edx, dword ptr[qary+4]

mov al, type w1
mov bl, lengthof w1
mov cl, sizeof w1

mov al, type bary1
mov bl, lengthof bary1
mov cl, sizeof bary1

mov al, type bary2
mov bl, lengthof bary2
mov cl, sizeof bary2
count=5
mov al,count
```

al	3
al	C
bx	03
ecx	0 2333
edx	57856
al	2
bl	2
cl	4
al	1
bl	9
cl	9
al	1
bl	7
cl	7
al	5

2

0370

National University of Computer and Emerging Sciences

Islamabad Campus

Question 3 [10 Marks]

Write an optimized code that copies data from **ary1** to **ary2** by swapping each byte in the **ary1**. Your final **ary2** should look as shown below

```
ary2 dw 3412H, 7856H, 0CDABH
```

NOTE: must use **XCHG** and **LOOP** instructions this will produce an optimized code. Do not use unconditional jumps

```
.data
      ary1 dw 1234H,5678H,0ABCDH
      ary2 dw 0, 0, 0
.code
   mov eax, OFFSET ary1
    mov ecx, 3
L1:  mov ebx, [ary2 + 6]
    XCHG ,(ary2[~~ecx~~], ary1[ecx])
      ~~Jez ecx, Jum~~
        Cmp ecx, 0 ; Check if ecx has
                            gone to zero if
        Je L2
L2:   Loop L1 ; loop to L1 as ecx    yes then  stop loop.
               decreases automatically
    invoke ExitProcess, 0
```

Question 4 [7+3=10 Marks]

Fill memory and update the given **memory** after executing the following code on data given below. **NOTE:** No need to convert characters to **ASCII** use them as characters. Watch your updated memory carefully.

```
.data

 string1 db 'eman srotcurtsni ruoy elcric sunob erocs ot'
 string2 db $-string1 dup(0)
 stp dd string1
```

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0100	e	m	a	n	s	y	o	t	co	u	r	t	s	n	i	r
0110	v	o	y	e	l	c	r	i	c	s	v	n	o	b	e	r
0120	o	c	s	o	t	t	o	s	c	o	r	e	b	o	n	u
0130	s	c	i	r	c	l	e	y	o	~~u~~	~~r~~	i	n	s	t	r

u c t o r s n a ~~me~~

```
.code
   mov eax,0
   mov ebx,0
   mov eax,offset stp
   mov esi,[eax]
   mov ecx, sizeof string1
   L1:
        mov al,[esi+ebx]
        mov string2[ecx],al
        inc ebx
  LOOP L1
```

10

Question 5 [5+5+5+5=20 Marks]

I. Update the given flags after executing the following code? No marks for direct answer use rough space

```
mov al,+127
add al,1
```

OF	SIGN	ZF	AF	PF	CF
1	0	1	0	1	1

ROUGH WORK:

1

1111111

+1

1 00000000

0370

National University of Computer and Emerging Sciences

Islamabad Campus

II. Consider the following **data declaration** and fill in the given memory in hexadecimal (**h**).
NOTE: Highlight spaces create due to **align directive** and fill them with **zero**

```
.data
        var1 db 7,8,9
        align 4
        var2 dd 012345678H
        var3 db 2
        align 2
        var4 dw 5,6
        align dword//
        var5 db 7
        var6 dw 01234H
```

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0100	7	8	9	00	78	56	34	12	2	00	05	00	06	00	00	00
0110	7	34	12													
0120																
0130																

III. Write code that add **v1** and **v2** and save sum in result for **.386** architecture
HINT: ADC dest, source ; ADDS destination + source + Carry

```
.data
        v1 dq 0123456789ABCDEFH
        v2 dq 0123456FFABCDEFH
        result dq 0
```

.code

mov eax, V1

~~ADC~~

~~ADD eax, V2~~

ADC eax, V2, 6

mov result, eax

Row.

① ① ①①①①

12345678ABCDEF

123456FFABCDEF

BCDEFF

①①①①①①

12345678ABCDEF

123456FFABCDEF

2468AC79579BDE

1 111

1111

1111

111

0000

1 11

1110

1110

1100

0370

IV. Update the given flags after executing the following code? What happened to **Carry**, **Overflow** and **Auxiliary** flags. No marks for direct answer use rough space

```
mov bx,1DEFH
AND bx,5AA5H
ADD
```

OF	SIGN	ZF	AF	PF	CF
0	0	0	1	~~0~~	~~1~~

ROUGH WORK:

```
 ①①①
 1DEF          0111        13
 5AA5                      10
 78A5                      ~~23~~
                           ~~16~~
                           17
```

ROUGH SPACE