

Ayesha Mehmood

ID # 6832

Subject # Microprocessor

And Assembly
Language

Degree # BSCS

Semester # 8th

Final - Term

Date # 29 - June - 2020

①

Question # 5

Answer

AI : 40h

AX : Ax0038h

EDX : 3

Question # 8

Answer

(a) Push ebx

Push eax

Pop ebx

Pop eax

②

Before the Loop:

mov esi, OFFSET source ; offset of variable

mov ebx, 1 ; byte format

mov ecx, SIZEOF source ; counter

call DumpMem ; require the initial value of esi, ebx, ecx.

After the Loop:

mov esi, OFFSET target ; offset of variable

mov ebx, 1 ; byte format

mov ecx, SIZEOF target ; counter

call DumpMem ; required the initial value of esi, ebx, ecx

(3)

c) INCLUDE Irvine32.inc

• data

Count DWORD?

• code

main PROC

mov eax, 0 + (0 * 16)

mov ecx, 16

L1:

mov count, ecx

push eax

mov ecx, 16

(4) (16)

L2:

```
call setTextColor  
push eax  
mov al, 'H'  
call writeChar
```

```
pop eax  
inc eax
```

Loop L2

```
call crlf  
pop eax  
add eax, 16  
mov ecx, count
```

```
call crlf  
call waitmsg
```

```
exit
```

```
main ENDP
```

```
END main
```

5

Question # 06

Answer

Solution:

USE the AND instructions

(a)

mov al, 'a' ; al = 01100001b

and al, 11011111b ; al = 01000001b

(b)

mov al, b ; al = 00000110b

or al, 00110000b ; al = 00110110b

(c)

and al, 00001011b ; clear unwanted bits

cmp al, 00001011b ; check remaining bits

je L1 ; all set? jump to L1

6

Question # 03

Answer

mov eax, TYPE myBytes ; a. 1

mov eax, LENGTHOF myBytes ; b. 4

mov eax, SIZEOF myBytes ; c. 4

mov eax, TYPE myword ; d. 2

mov eax, LENGTHOF myword ; e. 4

mov eax, SIZEOF myword ; f. 8

mov eax, SIZEOF myString ; g. 5

7

Question # 01

Answer

MOVZX cx, bl. cx = 009Bh

MOVSX cx, bl. cx = 009Bh

XCHG val2, ax. val2 = 1000h

mov al, [array B+1]; (d) AL = 20h

mov ax, [array W+2]; (e) AX = 200h

mov eax, [array D+4]; (f) EAX = 20000h

8

Question # 07

Answer

a) mov eax, var1

cmp eax, var2

jle L1

mov var3, 110

mov var4, 90

jmp L2

L1: mov var3, ~~110~~ 128

~~jmp L2~~

c)

top: cmp eax, ebx ; Check loop
condition

(9)

```
jae next ; false? exit  
loop
```

```
inc eax ; body of loop
```

```
jmp top ; repeat the  
loop
```

next:

b)

```
cmp val1, ecx  
jna L1
```

```
cmp ecx, edx
```

```
jna L1
```

```
mov x, 30
```

```
jmp next
```

```
L1: mov x, 40
```

next:

10

Question # 02

Answer

MOV CX, 1
Sub CX, 1; (a) CX = 0 ZF = 1

MOV CX, 0
Sub CX, 1; (b) CX = -1 SF = 1

MOV AL, 0FFh
add AL, 1; (c) AL = 00 CF = 1

MOV AL, 0
Sub AL, 1; (d) AL = FF CF = 1

MOV AL, 7Fh
add AL, 1; (e) AL = 80 OF = 1

MOV AL, -128
neg AL; (f) CF = 1 OF = 1

11

Question # 04

Answer

• data

val32 LABEL DWORD

varB BYTE 78h, 56h, 34h, 12h

val8 LABEL BYTE

varD DWORD 12345678h

• code

mov bl, BYTE PTR varD; (a) BL = 78h

mov eax, DWORD PTR varB; (b)

EAX = 78563412h

mov al, val8; (c) AL = 78h

mov eax, val32; (d) EAX = 12345678h