

IQRA NATIONAL UNIVERSITY

PESHAWAR

SUBMITTED BY SHARIQ

ID

13698

SUBJECT

ASSEMBLY

LAUNGAUGE

Q1 :-

Answer :-

(a) $EAX = 0000ABCDH$

(b) $EAX = FFFFABCDH$

(c) $AX = 3000h$

(d) $AX = 7000h$

(e) $val 2 = 3000h$

(f) $val 1 = 3000h$

(g) $AI = 50$

(h) $AI = 10$

(i) $AX = 3000h$

(j) $AX = 1000h$

(k) $EAX = 30000000h$

(l) $EAX = 20000000h$

Q2:-

Answer:-

mov ax, 7FF0h

add al, 10h ; (a) CF=1, SF=0, ZF=1, OF=0

add ah, 1 ; (b) CF=0 SF=1, ZF=0, OF=1

add ax, 2 ; (c) CF=0, SF=1, ZF=0, OF=0

mov al, 1

sub al, 2 ; (d) CF=0, SF=0, ZF=1, OF=1

Q3 :-

Part (1)

(a) $EAX = 1$

(b) $EAX = 6$

(c) $EAX = 6$

(d) $EAX = 2$

(e) $EAX = 32$

(f) $EAX = 64$

(g) $EAX = 18$

Q3:-

(b)

```
mov eax, DWORD PTR List B
```

(c)

```
List B LABEL DWORD
```

```
List B WORD 3 DUP (?) 2000h
```

```
• data
```

```
mov eax, List B.
```

Q4:-

Ans:-

mov esi OFFSET myBytes

mov al, [esi] ; a. AL = 10h

mov al, [esi+3] ; b. AL = 40h

mov esi, OFFSET mywords+2 ;

mov ax, [esi] ; c. AX = 003Bh

mov edi, 8

mov edx, [mydoubles+edi] ; d. EDX = 3

mov edx, myDoubles[edi] ; e. EDX = 3

mov ebx, myPointer

mov eax, [ebx+4] ; f. Eax = 2

Q5:

①

mov eax, var 1

cmp eax, var 2

jle L1

mov var 3, 10

mov var 4, 30

jmp L2

L1: mov var 3, 15

L2:

②

cmp val 1, ecx

jna else

cmp ecx, edx

jna else

mov A, 12

jmp next

else: mov B, 6

next:

Q5 :- Part (3)

while (ebx < eax)

ebx = ebx + 1

Answer :-

top: cmp ebx, eax

jae next

inc ebx

jmb top

next;

Q6:-

Part ①:-

mov eax, 0

mov ecx, 10 ; outerloop counter

L1: mov eax, 3

mov ecx, 5 innerloop counter

L2: add eax, 5

LOOP L2 repeat inner loop

LOOP L1 repeat outerloop.

Answer:-

Infinite loop.

Q6 :-

Part 2.

• 386

• model flat, stdcall

• Stack 4096

ExitProcess PROTO, dwExitCode : DWORD

• Code

main PROC

mov eax, 3h

mov ebx, 8h

mov ecx, 1h

mov edx, 8h

add eax, ebx

add ecx, edx

sub eax, ecx

INVOKE ExitProcess, 0

main ENDP

END main.

Q6:

Part (3)

• model-flat, stdcall

• stack 4096

ExitProcess PROTO, dwExitCode: Dword

• data

array Word 0, 2, 5, 9, 10

new Array DWORD LENGTHOF array DUP(?)

• Code

main PROC

mov ecx, LENGTHOF array

mov esi, OFFSET array

mov edi, OFFSET new Array

L1:

mov eax, 0

mov ax, [esi]

mov [edi], eax

add esi, TYPE array

add edi, TYPE new Array

loop L1

invoke ExitProcess, 0

main ENDP

END main

Q6: Part 4:-

```
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
```

```
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
```

```
LOOP L1
```

```
call crlf
```

```
call waitMSG
```

```
exit
```

```
main ENDP
```