National University of Computer and Emerging Sciences, Lahore Campus

Course Name:	Computer Organization and Assembly Language	Course Code:	EE213
Program:	BS(Computer Science)	Semester:	Fall 2017
Duration:	3 Hours	Total Marks:	90
Paper Date:	26th December, 2017	Weight	45%
Section:	ALL	Page(s):	10
Exam Type:	Final		

Student : Name:_____ **Roll No.**_____ **Section:**_____

Instruction/Notes:

1. Exam is Open book, Open notes.
2. Properly comment your code.
3. You need not to copy code from book. Just call any subroutine needed .
4. Write your answer in the space provided. You **can take extra sheets BUT they WONT BE ATTACHED WITH THE QUESTION PAPER OR MARKED.**

Question 1: Short Questions. [30 Marks]

(no cutting, no over-writing , no partial marking)

1. How many times int 0x01 will come in the following code? **(3 Marks)**

```
pushf                   ; setting TF=1
pop ax
or ax, 0x100
push ax
popf

mov cx, 5
rep stosb
mov ax, 0x4c00
int 21h
```

Solution:

7 times

2. Suppose initpcb code is hooked on interrupt 0x80. This isr, along with timer and all variables are made tsr. When int 0x80 is called from another program, what will be the value of ss in pcb variable? Currently initpcb initializes it on DS. **(3 Marks)**

CS

3. Suppose mytask code in example runs a finite number of times and the loop ends. The code finally reaches the return statement at line 83 and performs ret 2. The return address it pops is 0xFA20.
Where do you think, in complete code given in book, should this address be placed on stack for mytask? (Mention line number) **(3 Marks)**

After line 115 with this mov [si], 0xFA20 statement.

4. Int 0x03 is responsible for: **(2 Marks)**

(a) Inserting break point
(b) Removing breakpoint

(c) Removing opcode
(d) Restoring opcode

5. Write code of two ways through which timer interrupt can be disabled. **(2+2 Marks)**

```
In al, 0x21

Or al, 0x01

Out 0x21, al
```

```
cli
```

6. In an inter segment call, what will be the order in which segment and offset of return address will be placed on the stack. **(2 Marks)**

Segment at higher address, offset at lower

7. Write code of two ways through which we can read scan code of a key board button. **(2+2 Marks)**

```
In al, 0x60
```

```
Mov ah, 0

Int 0x16

;ah will have scan code
```

8. What is the maximum size of IVT? (mention in kb) **(1 Mark)**

1kb

9. Which of the following is a syntax error? **(1 Mark)**

a. mov ax, [bx-5*7/2]
b. mov ax, [bx+si-5]
c. mov ax, [bx-si +5]

10. Convert the following expressions into assembly language code using only shift, addition, and subtraction instructions in place of the multiplication: **(2+2 Marks)**

a. AX * 15

```
Sub bx, bx
Add bx, ax
Shl ax, 4
Sub ax, bx
```

b. AX * 129

```
Sub bx, bx
Add bx, ax
Shl ax, 7
add ax, bx
```

11. What will be the value of **CX**, **DI** and **ZF** after the following code? If **DI** at start of code is 0x0111.**(3 Marks)**

```
[org 0x100]
push ds
pop es
mov al, [string]
mov cx, 5
mov di, string
cld
rep scasb

mov ax, 0x4c00
int 21h
string: db 'hello',0
```

CX___3_____

DI___0113_____

ZF_____0_____

Question 2: (30 Marks)

Read the following problem carefully and write complete code.
Consider we are implementing a **small message display** service. The service performs two functions: display user messages on video screen and secondly facilitate the user to delete any previously displayed message. The user can never send a message longer than 80 characters. Longer messages sent are automatically discarded by the service. After displaying the message the service facilitates the user to delete any previous message, already displayed on the screen within 10 seconds after it was sent. The service displays the string "*this message has been deleted"* in the deleted message's space (just like whatsapp)
Suppose that the above service uses interrupts interface to export its services.
For your simplicity of implementation, let's divide the service's functionalities in two interrupts:
Int 0x60: It displays the messages on video screen and takes the following parameters:
ES: segment where string is present in data segment
DI: offset of string
DH: row number of the video screen where to display
(You don't need to take input of messages from user. Imagine that messages are already present in the data segment and you pass only the segment and offset of that string to int 0x60).

Int 0x65: It deletes the message from the video screen and takes the following parameters:
DH: row number from where the message is to be deleted.
Some important conditions: there is no restriction of over writing a message. The user can call int 0x60 and over-write any previous message as well. You can simply call subroutines like printstring, strlen from book rather than writing their code.

25 times dw ● - 1
time : ~~dw 0~~

```
int 60 isr:
    sti        ; so timer ISR can come

→ es has segment
→ di has offset
→ DH has row no.

mov   bx, dh      ; bx will have index of
mov   bh, 0         time array

mov time[bx], 0

; call strlen to calculate length of message
; if length of string > 80 → exit
; clear row & put space.
else

push row
push col
push length
call printstr.
cli
mov   time[bx], 0
sti
```

2	① calculate length
2	② print string
3	③ time Note
3	④ enable interrupt

(10)

msg: "this message has been deleted."

int 66isr:

→ mov bl, dh ; for times away
→ mov bh, 0

cmp time[bx], 180
jl exit

~~row~~ si
mov ~~ax~~, 80
mul dh
shl ~~si~~, 1

mov cx, 80
~~mov si, ax~~

l1: mov [si], 0x0720
add si, 2
loop l1

push bx
push 0
push strlen
call printstr

3 ① check time
2 ② clear line
3 ③ print string
2 ④ set time to -1

mov time[bx], -1

(10)

Department of Computer Science Page 5 of 10

```
timer:
    mov  cx, 25
    mov  si, 0
l1: cmp  time[si], -1
    je   skip
    add  time[si], 1
skip: add si, 2
    loop l1
```

3 ① time array & its default value

② other variables msg

3 ③ interrupt time

Main

; hooking interrupts

(3)

(10)

Department of Computer Science Page 6 of 10

are created and are running. Each process runs for 5 seconds and then stops, staying in the list as a dormant process but not running. When all processes have stopped, a message is displayed on the screen "multi-tasking program ending..." and after 5 seconds, the program terminates normally and is completely removed from memory.

Suggest all changes required in the code to achieve the above, clearly mentioning the line numbers where you want to make changes/insert code. **(30 Marks)**

; change in initpcb. ; initialize dummy to 91

(anywhere below 100 to 117)

```
mov [pcb + bx + 30], 91
```

Change in timer

variables :-

```
flag : db 0          ; to check if all processes have ended
count: db 0          ; to count for 5 sec after all process end.
newtick: db 54 (/55) ; to create new process after every 3 sec.
```

code:

line 134 :

① to check if all processes have already ended

```
cmp byte [flag], 1
jne cont1
inc byte [end], 1
; EOI
iret
```


Department of Computer Science Page 7 of 10

(line 134) Cont1: push ds

⋮

After line 138:

② create new process in timer after every 3 seconds

```
dec   byte [newtick]
cmp   byte [newtick], 0
jne   cont2
push  cs
mov   bx, mytask
push  bx
push  word [lineno]
call  initpcb
inc   word [lineno]

mov   byte [newtick], 54
```

(line 140)
cont2:

⋮

till line 145 same code.

Department of Computer Science Page 8 of 10

after line 145:

③ decrement execution time of processes other than main

```
cmp   word[current], 0
je    cont3
dec   [pcb + bx + 30], 1
cmp   [pcb + bx + 30], 0
jne   cont3
mov   [pcb + bx + 30], -1
inc   byte [count], 1
```

(line 146)

cont3 :

(same code)

(till line 167)

at line 167 :-

l1 :

④ find next process whose duty does not have -1

```
mov   bx, [pcb + bx + 28]
mov   [current], bx
mov   cl, 5
shl   bx, cl
cmp   [pcb + bx + 30], -1
je    l1
```

(continue from line 172)

Department of Computer Science

Page 9 of 10

Changes in main :-

after line 203 :-

```
204 l2: cmp   byte [count], 31
        jne   l2

        mov   byte[flag], 1

    l3: cmp   byte [end], 91
        jne   l3
```

→ unhook time, printstring "_____"

```
mov   ax, 0x4c00

int   21h.
```

---

① in initpcb (4)

② changes in main (6)

③ timer :-

i) check if all process have ended (5)

ii) create new process (5)

iii) dec exe time & set dly to -1 & inc count (5)

iv) find next process (5)

~Best of Luck! ☺~