|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Computer Organization and Assembly Language** | **Course Code:** | **EE229** |
| **Program:** | **BS(Computer Science)** | **Semester:** | **Fall 2019** |
| **Duration:** | **60 Minutes** | **Total Marks:** | **22** |
| **Paper Date:** | **09-Nov-2019** | **Page(s):** | **5** |
| **Section:** | **ALL** | **Section:** |  |
| **Exam:** | **Midterm Exam 2** | **Roll No:** |  |
| **Instruction/Notes:** | 1. Answer in the space provided 2. You can ask for rough sheets but **they will not be graded or marked** 3. In case of confusion or ambiguity make a reasonable assumption. 4. Questions are not allowed 5. This is open book and open notes paper  Good luck! | | |

Q1: (5 Points) Following program has a function add1 that takes 2 numbers from stack and if sum of these two number is greater than, 0 it return their sum through stack otherwise it return 0 through stack. Code has some logical errors correct those errors so that you can pop correct answer in dx register. You can add or modify existing lines but you cannot remove any line.

;; Rewrite Your code here

[org 0x100]

jmp start

add1:

push bp

mov bp,sp

sub sp,2 ;; local variable

mov sp , [bp + 2]

mov [bp - 2], sp

mov ax , [bp + 4]

add [bp - 2 ] , ax

mov ax, [bp -2]

mov [bp + 8] ,ax

cmp word [bp + 8] , 0

ja end

mov word [bp + 8], 0

end:

pop bp

ret 2

start:

mov ax,5

push ax

mov bx, 2

push bx

call add1

pop dx ;; value of DX should be 7

mov ax, 4c00h

Int 21h

; your code ends here

mov ax, 0x4c00

int 0x21

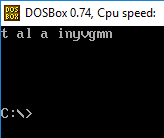
Q2: (10 Points) Display encoding string on screen

Write a function that takes an input string and prints the encoding of that string on screen. Assume that the input string will only contain characters a-z and space. Following is the description of encoding.

Encoding means mapping each character to a new value. The mapping will be given in form of two string, **Original** and **Mapped**. The two arrays show how to map a-z characters to new ones. For example **j** should be mapped to **f** while encoding, **o** to **w**, **q** to **p** and so on…

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Original :db | " | j | o | q | s | g | v | b | m | f | k | t | p | i | e | c | h |  | u | x | n | r | d | a | l | w | z | y | " |
| Mapped : db | " | f | w | p | i | j | x | b | l | u | r | n | z | t | g | q | e |  | y | c | m | h | v | a | s | o | d | k | " |

(You can assume there will be no spaces in Original and Mapped and their length is 26. The above spaces are just for readability)

You function will take one string as input and will print its encoding on screen

For example

if input string= “i am a student”

Encoded string to be displayed on screen will be “t al a inyvgmn”

Note that **i** is mapped to **t**, **a** to **a**, **m** to **l** and so on … spaces are not changed

Figure 1: Expected Output for Original, Mapped and input String given in question statement

**Rules of writing code:**

Your code should only use string instructions while working with strings, here **input\_string**, **Original** and **Mapped** are all strings.

The above given values of **original** and **mapped** are just one example, your code should work with and shuffled order of characters.

You can used helper function of clrscr and strlen,, do not write code of these helper functions on your answer.

|  |
| --- |
| **;; Wri****te your code in provided template**  [org 0x100]  jmp start  **Original**:db "ptncoflyrqebsgkjwadzxmhviu";  **Mapped** :db "aenglbstfihpqvkcmudrjxoywz" ;  **input\_string**: db ‘i am a student of coal ’,0  ; string to be displayed on screen in encoded format on screen  Figure 2 Expected output for given Original, Mapped and Input\_String in this code template  ; assume clrscrn function is given  ;assume strlen function is given  **printEncodedString**:  ;write your function here  ; more space for question 2  start:  push input\_string  call printEncodedString  mov ax, 4c00h  Int 21h |

Q3: (2 Points) Software Interrupts

Which interrupt will be hooked after execution of following code?

|  |
| --- |
| [org 0x100]  ;;;; myISR is written here  xor ax,ax  mov es, ax  mov [es: 0x110 ] , myISR  mov [es : 0x112] , CS  mov ax, 4c00h  Int 21h |
| **Interrupt Number :** |

Q4: (5 Points) Software Interrupts

Write a program that takes a positive integer as input from user using int 16h and stores it in DX register. int 16h takes only one character but your program should takes input from user until user presses enter key. No need to hook int 16h. ASCII value of enter key is 13 in decimal.

Assume that user will enter only digits and your number will not exceed form the maximum limit of DX register.

For example:

When program executes it will wait for input, then user enters 1, 2, 3 and enter key. As soon as enter is pressed program will terminate and value of DX will be 123 (7Bh). Assume that user will always enter one or more digits (0-9) before pressing enter key and the entered integer will be in range of DX. Refer figure 3.

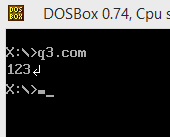
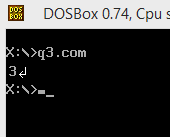
 

Figure 3 Example 1: DX should be 7Bh or 123 decimal before termination

Figure 4 Example 2: DX should be 3 before termination

|  |
| --- |
| **Following 2 lines generate an interrupt of get a character**  mov AH, 0  int 16hh  wait for a key press and when you entered a key you will get:   1. AH = keyboard scan code 2. AL = ASCII value of input character or zero if special function key |

|  |  |
| --- | --- |
| [org 0x100]  ; write code of Q4 here  mov ax, 4c00h  Int 21h |  |