National University of Computer and Emerging Sciences 

**Laboratory Manual**

*for*

**Computer Organization and Assembly Language Programming**

| Course Instructor | Aleena Ahmed |
| --- | --- |
| Lab Instructor | Sana Ejaz |
| Semester | Fall 2024 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

Page 1

**OBJECTIVES:**

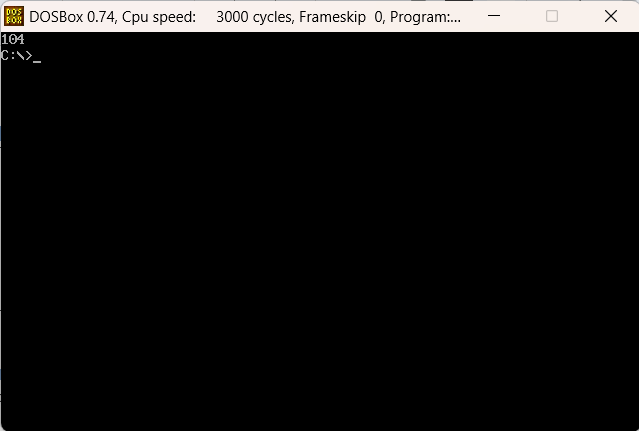
∙ Understand the purpose and functionality of software interrupts.

∙ Learn how to use different software interrupts for input/output operations. ∙ Gain hands-on experience with common DOS interrupts to manipulate hardware and software resources.

**Instructions:**

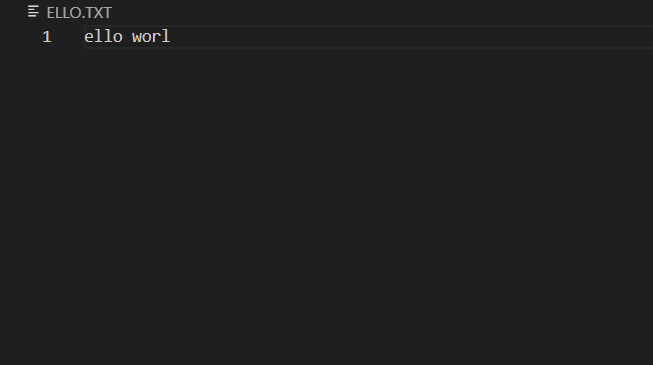
| **1. Submit work in a single Word file with screenshots of meaningful results. 2. Make proper subroutines**  **3. Use Delay Function (if required)** |
| --- |

**Task 1: Create a program that takes a single key input from the user using INT 16h and then displays the ASCII code of the entered key using INT 21h.**

****

| [org 0x0100]  JMP start  clrscrn:  PUSH AX  MOV AX, 0x0003  INT 0x10  POP AX  RET  printnum:  PUSH BP  MOV BP, SP  PUSHA  XOR AX, AX  MOV AL, [BP + 4]  MOV BX, 10  MOV CX, 0  takedigit:  MOV DX, 0  DIV BX  MOV DH, 0x07  ADD DX, 0x30  PUSH DX  INC CX  OR AX, AX  JNZ takedigit  MOV AH, 0x06  prnt:  POP DX  INT 0x21  LOOP prnt  POPA  MOV SP, BP  POP BP  RET 2  takeKey:  PUSHA  CALL clrscrn  XOR AX, AX  INT 0x16  PUSH AX  CALL printnum  POPA  RET  start:  CALL takeKey    MOV AX, 0x4C00  INT 0x21 |
| --- |

**Task 2: Write an assembly program that creates a text file, writes a message into it, and then closes the file using INT 21h.**

****

| [org 0x0100]  JMP start  filename: db 'ello.txt', 0  string: db 'ello worl'  size: dw size - string  fileHandling:  PUSHA  MOV AH, 0x3C  XOR CX, CX  MOV DX, filename  INT 0x21  JC error    MOV BX, AX  MOV AH, 0x40  MOV CX, [size]  MOV DX, string  INT 0x21  JC error  JMP terminate  error:  MOV AX, 0x4C01  INT 0x21  terminate:  POPA  RET  start:  CALL fileHandling  MOV AX, 0x4C00  INT 0x21 |
| --- |

**Task 3: Write an assembly language program that takes a message from the user using INT 21h and then displays the message on the screen. Experiment with different function calls for printing the string.**

| [org 0x0100]  JMP start  inputStr: db 'Input String: $'  outputStr: db 'Output String: $'  bufferSize: db 0xFF  inputLength: db 0  buffer: times 500 db 0  takeInput:  PUSHA  MOV AH, 0x09  MOV DX, inputStr  INT 0x21  MOV AH, 0x0A  MOV DX, bufferSize  INT 0x21  MOV AH, 0x02  MOV DL, 10  INT 0x21    MOV DL, 13  INT 0x21  XOR BX, BX  MOV BL, [inputLength]  MOV word [buffer + BX], '$'  MOV AH, 0x09  MOV DX, outputStr  INT 0x21  MOV DX, buffer  INT 0x21  POPA  RET  start:  CALL takeInput  MOV AX, 0x4C00  INT 0x21 |
| --- |

