|  |
| --- |
| FAST National University |
| **Lab 7** |
|  |

**Computer Organization and Assembly Language**

|  |  |
| --- | --- |
| **Student Name** | Moeez Ali |
| **Registration #** | 21L-1798 |
| **Instructor** | Hazoor Ahmad |
| **Class** | CS3 |
| **Section** | D |
| **Semester** | Fall 2022 |

Fast School of Computing

FAST-NU, Lahore, Pakistan

# Activity 1

## **Assembly Language Code**

[org 0x100]

jmp start

roll: db 1,7,9,8

a: db 0

b: db 0

c: db 0

d: db 0

Mr1: db 0,0

Mr2: db 0,0

Nr1: db 0,0

Nr2: db 0,0

Or1: db 0,0

Or2: db 0,0

loadmat:

push bp

mov bp,sp

push ax

push bx

push cx

mov ax,[bp+4]

mov [Nr1+1],ax

mov ax,[bp+6]

mov [Nr1],ax

mov ax,[bp+8]

mov [Mr1+1],ax

mov ax,[bp+10]

mov [Mr1],ax

mov ax,[bp+4]

mov [Mr2+1],ax

mov ax,[bp+6]

mov [Mr2],ax

mov ax,[bp+8]

mov [Nr2+1],ax

mov ax,[bp+10]

mov [Nr2],ax

pop cx

pop bx

pop ax

pop bp

ret 4

matmul:

push bp

mov bp,sp

push ax

push bx

push cx

mov ax,[Mr1]

mov bx,[Nr1]

mul bx

mov dx,bx

mov ax,[Mr2]

mov bx,[Nr2]

mul bx

add dx,bx

mov [Or1],dx

mov ax,[Or1]

push ax

call print

mov ax,[Mr1]

mov bx,[Nr1+1]

mul bx

mov dx,bx

mov ax,[Mr2+1]

mov bx,[Nr2]

mul bx

add dx,bx

mov [Or1+1],dx

mov ax,[Or1+1]

push ax

call print

mov ax,[Mr1+1]

mov bx,[Nr1]

mul bx

mov dx,bx

mov ax,[Mr2+1]

mov bx,[Nr2]

mul bx

add dx,bx

mov [Or2],dx

mov ax,[Or2]

push ax

call print

mov ax,[Mr1+1]

mov bx,[Nr1+1]

mul bx

mov dx,bx

mov ax,[Mr2+1]

mov bx,[Nr2+1]

mul bx

add dx,bx

mov [Or2+1],dx

mov ax,[Or2+1]

push ax

call print

pop cx

pop bx

pop ax

pop bp

ret

print:

push bp

mov bp, sp

push es

push ax

push bx

push cx

push dx

push di

mov ax, 0xb800

mov es, ax

mov ax, [bp+4]

mov bx, 10

mov cx, 0

nextdigit:

mov dx, 0

div bx

add dl, 0x30

push dx

inc cx

cmp ax, 0

jnz nextdigit

mov di, 0

nextpos:

pop dx

mov dh, 0x07

mov [es:di], dx

add di, 2

loop nextpos

pop di

pop dx

pop cx

pop bx

pop ax

pop es

pop bp

ret 2

start:

mov si,0

mov ax,[roll]

push ax

mov ax,[roll+1]

push ax

mov ax,[roll+2]

push ax

mov ax,[roll+3]

push ax

call loadmat

call matmul

mov ax,0x4c00

int 0x21

## **Debugging Screenshots**

