

# **Manas Pratim Biswas**

## **IT-A1-025 ASM Lab Assignment 1**

---

- 1. Write an Assembly Language Program to add two two-digit integers and store the result in the DX register.**

```
.model small
.stack 100h
.data
.code
main proc
    mov bx,5
    mov dx,3
    add dx,bx
    add dx,48
    mov ah,2
    int 21h
    mov ah,4ch
    int 21h
main endp
end main
```

- 
- 2. Write a program to compute the difference of two numbers and store the magnitude and sign in memory.**

```
.model small
.stack 100h
.data
.code
main proc
    mov bl,9
    mov dl,12
    sub dl, bl
    add dl,48
    mov ah,2
```

```

        int 21h
        mov ah, 4ch
        int 21h
main endp
end main

```

---

**3. Write an ALP, which will prompt you to enter your name as a string, and display the inputted string.**

```

.model small
.stack 100h
.data
    arr db 20 DUP('$')
    msg db 'Enter you name : $'
.code
main proc
    mov ax,@data
    mov ds,ax
    mov ax,offset msg
    mov dx,ax
    mov ah,9
    int 21h
    mov si, offset arr
lp:
    mov ah,1
    int 21h
    cmp al,13
    je progend
    mov [si],al
    inc si
    jmp lp
progend :
    mov dx, offset arr
    mov ah,9
    int 21h
    mov ah,4ch
    int 21h

```

```
main endp
end main
```

---

**4. Write an ALP to check the length of a given string.**

```
.model small
.stack 100h
.data
    arr db 20 DUP('$')
    msg db 'Enter a string : $'
    msgg db 'Length of the entered string = $'
.code
main proc
    mov ax,@data
    mov ds,ax
    mov ax,offset msg
    mov dx,ax
    mov ah,9
    int 21h
    mov cx,0
    mov si, offset arr
lp:
    mov ah,1
    int 21h
    cmp al,13
    je progend
    mov [si],al
    inc cx
    inc si
    jmp lp
progend :
    mov dx, offset msgg
    mov ah,9
    int 21h
    mov dx,cx
    add dx,48
    mov ah,2
```

```
    int 21h
    mov ah,4ch
    int 21h
main endp
end main
```

---

**5. Write an ALP which will display a two-digit number that will be taken from the keyboard.**

```
.model small
.stack 100h
.data
    arr db 20 DUP('$')
    msg db 'Enter a two digit number : $'
    msgg db 'The number entered = $'
.code
main proc
    mov ax,@data
    mov ds,ax
    mov ax,offset msg
    mov dx,ax
    mov ah,9
    int 21h
    mov si, offset arr
lp:
    mov ah,1
    int 21h
    cmp al,13
    je progend
    mov [si],al
    inc si
    jmp lp
progend :
    mov dx, offset msgg
    mov ah,9
    int 21h
    mov dx,offset arr
```

```
    mov ah,9
    int 21h
    mov ah,4ch
    int 21h
main endp
end main
```

---