

Assignment 1

## 1.1 Write a program to find the addition of two bytes.

```

.model small
.stack 64h
.data
arr1 db 10h,20h,30h,40h,50h
arr2 db 30h,20h,40h,40h,0f0h
count dw 05h
arr3 db 6 dup<00> ;6 is the size of resultant array initialised with 00

.code
main proc near
    mov ax,@data
    mov ds,ax
    mov cx,count
    clc
    LEA BX,arr1          ;load effective address
    LEA SI,arr2          ;SI - Source Index
    LEA DI,arr3          ;DI - Destination Index

    L1: mov AL,[BX]
        adc AL,[SI]      ;ADC Add with carry
        mov [DI],AL      ;AL contains addition
        inc SI           ;Increment all the pointers
        inc DI
        inc BX
        loop L1

    mov AL,00h
    adc AL,00h
    mov [DI],AL          ;To transfer higher byte to resultant array

    mov ah,04ch
    int 21h
    main endp
end main

```

## 1.2 Write a program to find the occurrence of the given number into the array.

```

.model small
.stack 64h
.data
arr1 db 10h,30h,30h,40h,50h
arrsize dw 05h
occurcount db 00h
find db 30h           ;The number whose occurrence to be found in the array

.code
main proc near
    mov AX,@data
    mov DS,AX
    mov CX,arrsize
    clc
    LEA BX,arr1        ;load effective address

```

```

    L1: mov AL,[BX]
        cmp AL,find
        jne skip
        inc occurcount

        skip:
        inc BX
        loop L1

    mov AL,00h
    mov AH,04ch
    int 21h
    main endp
end main

```

1.3 Write a program to sort the numbers in the given array in the descending order.

```

.model small
.stack 64h
.data
arr1 db 20h,25h,10h,50h,55h
outsidecounter dw 04h           ;Total size of array - 1
insidecounter dw 00h
tempcounter dw 00h

.code
main proc near
    mov ax,@data
    mov ds,ax
    mov cx,outsidecounter
    CLC                     ;Clear Carry
    LEA BX,arr1             ;Load effective address
    OUTERLOOP:
        mov insidecounter, CX
        PUSH CX
        mov CX,insidecounter
        lea SI,arr1

        INSIDELOOP:
            mov DL,[SI]
            cmp DL,[SI+1]
            JG SKIP
            XCHG DL,[SI+1]
            MOV [SI],DL

            SKIP:
            INC SI

        loop INSIDELOOP
        POP CX
        loop OUTERLOOP
    mov ah,04ch
    int 21h
    main endp
end main

```

Assignment 2

2.1 Write a program to find the length of the string.

```
.model small
.stack 64h
.data
s1 DB 'The M.S.University of Baroda','$'
strlen DB ?

.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax
    CLD                                ;Increment DI after each character

    MOV AL,'$'
    MOV CX, 00FFH
    LEA DI,s1
    MOV BX,DI
    REPNE SCASB                        ;Scan string for NUL, decrementing CX for each char
                                        ;SCASB - Scan a Byte String
                                        ;REPNE - Repeat while not equal
;Scans the string searching for the first string element which is equal to the
value in the AL
    JNE SKIP
    DEC DI
    SKIP:SUB DI, BX
    MOV CX,DI
    MOV strlen,CL

    MOV AH,4Ch
    int 21h
    main endp
end main
```

2.2 Write a program to sort the occurrence of the given sub-string from the given string.

```
.model small
.stack 64h
.data
s1 DB 'MISSISSIPPI','$'
s2 DB 'SS','$'
len1 DB 0Ch
len2 DB 03h
count db 00h

.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax

    ;count no of iteration in CX
    mov AL,len1
    mov BL,len2
    sub AL,BL
    inc AL
    mov CL,AL    ;len1 - len2 + 1
    mov CH,00h

    CLD
    LEA bx,s1

;compare sub string and advance pointer by 1
l1:
    lea DI,s2
    mov SI,BX
    push CX
    mov CL,len2
    mov CH,00h

    REPE CMPSB        ;repeat while equal - compare string byte-wise
    cmp CX,0000h
    jne skip
    inc count
skip:
    inc BX
    pop CX
    loop l1

    mov ah,04ch
    int 21h
main endp
end main
```

Assignment 3

3.1 Write a program to perform multi byte multiplication.

```
.model small
.stack 64h
.data
md dw 1112h,2223h,3334h,4445h
mr dw 0001h
n db 04h
res dw 5 dup<0000h>

.code
main proc near
    mov ax, @data      ; Initialize data section
    mov ds, ax
    mov bx, 0000h

    mov CL,n           ;multibyte size
    mov CH,00h

    LEA SI,md           ;multiplicand
    LEA DI,res

    mov AX,CX
    add AX,AX
    add DI,AX

    sub AX,02h
    add SI,AX

l1:  mov AX,[SI]
     mul mr              ;multiplicant
     add AX,BX
     mov [DI],AX
     jnc skip
     inc DX

    skip:
     mov BX,DX
     sub SI,02h
     sub DI,02h
     loop l1

    mov [DI],BX

    mov ah, 4Ch         ; Terminate Program
    int 21h

    main endp
end main
```

### 3.2 Write a program to repeat or insert the given pattern into string.

```
.model small
.stack 64h
.data
len DB 05h
pattern DB '***++'
array DB 30 DUP<00>

.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax

    mov CX,06h    ;number of times we want to copy the given pattern

    LEA SI,pattern    ;source string or pattern to be copied
    LEA DI,array      ;destination array location

    L1:    push CX
           mov CX,len
           REP movsb    ;Repeate while Move string byte
           pop CX
           loop l1

    mov ah,04ch
    int 21h
    main endp
end main
```

Assignment 4

4.1 Write a program to enter the string from the keyboard and print it in the centre of the screen.

```
cur1 MACRO
    MOV AH,02
    INT 10h
ENDM

center MACRO
    ;MOV ZX,act_len
    ;MOV kb_name[BX],07
    mov bl,act_len
    mov bh,00h
    MOV kb_name[BX+1],'$'
    MOV DL, act_len
    SHR DL,01
    NEG DL
    ADD DL,40    ;DL = Column
    MOV DH,12    ;Row Number
    MOV BH,00    ;Page No
    cur1
ENDM

.model small
.stack 64h
.data
para1 label Byte
maxlen DB 30
act_len DB ?
kb_name DB 31 DUP (' ')
prompt DB 'Enter String : ','$' ;Prompt to the user to enter the string
.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax

    MOV AH,09h
    LEA DX,PROMPT
    INT 21h

    MOV AH,0Ah
    LEA DX, para1
    INT 21h

    CMP act_len,00
    JE Exit

    center
    MOV AH,09h
    LEA DX,kb_name
    INT 21h
Exit:    MOV AH,4Ch
        INT 21h
        END
```

4.2 Write a program to enter the string from the keyboard and print it in the centre of the screen using 01h interrupt.

```
cur1 MACRO
    MOV AH,02
    INT 10h
ENDM
```

```
center MACRO
    ;MOV ZX,act_len
    ;MOV kb_name[BX],07
    mov bl,act_len
    mov bh,00h
    MOV kb_name[BX+1],'$'
    MOV DL, act_len
    SHR DL,01
    NEG DL
    ADD DL,40    ;DL = Column
    MOV DH,12    ;Row Number
    MOV BH,00    ;Page No
    cur1
ENDM
```

```
.model small
.stack 64h
.data
para1 label Byte
maxlen DB 30
act_len DB ?
kb_name DB 31 DUP (' ')
prompt DB 'Enter String Character by Character : ','$'
        ;Prompt to the user to enter the character
s1 db 00
```

;Take Input string from the user and display it on the center of the screen using 01h interrupt

```
.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax
    LEA DI,s1

    MOV AH,09h
    LEA DX,PROMPT
    INT 21h
```



```
        L1: MOV AH,01h
           INT 21h
           MOV [DI],AL
           INC DI
           CMP AL,0dh  ;0dh = '$'
           JNE L1

MOV [DI], '$'

center
MOV AH,09h
LEA DX,s1
INT 21h

MOV AH,4Ch
INT 21h
END

        main endp
end main
```

Assignment 5

5.1 Write a program to display date and time of the system.

DIGITS MACRO data

```

MOV AL,data
SUB AH,AH    ;AH=00
MOV BL,10

                ;Divide AX to 10
DIV BL        ;Ten's place and One's place
PUSH AX       ;AL = Quotiant
DISP AL
POP AX
DISP AH       ;AH = Remainder One's Place

```

ENDM

YEARDISP MACRO data

```

MOV AX,data
MOV BX,1000

                ;Divide AX to 1000
DIV BX         ;Thousand's place and One's place
PUSH AX        ;AL = Quotiant
DISP AL
POP AX

MOV AL,AH
MOV BX,100
DIV BX
PUSH AX
DISP AL
POP AX

MOV AL,AH
MOV BX,10
DIV BX
PUSH AX
DISP AL
POP AX

DISP AH        ;AH = Remainder One's Place

```

ENDM

DISP MACRO dig

```

MOV AL,dig
ADD AL,30h    ;Hex to ASCII
MOV DL,AL
MOV AH,02h    ;Print Digit
INT 21h

```

ENDM

```

.model small
.stack 64h
.data

```

```
TMSG DB 'The time is : $'
```

```
DTSB DB '      Todays Date is : $'
```

```
SUN DB 'SUNDAY $'
```

```
MON DB 'MONDAY $'
```

```
TUE DB 'TUESDAY $'
```

```
WED DB 'WEDNESDAY $'
```

```
THUR DB 'THURSDAY $'
```

```
FRI DB 'FRIDAY $'
```

```
SAT DB 'SATURDAY $'
```

```
DAYTAG DW SUN,MON,TUE,WED,THUR,FRI,SAT
```

```
JAN DB 'JANUARY $'
```

```
FEB DB 'FEBRUARY $'
```

```
MAR DB 'MARCH $'
```

```
APR DB 'APRIL $'
```

```
MAY DB 'MAY $'
```

```
JUN DB 'JUNE $'
```

```
JUL DB 'JULY $'
```

```
AUG DB 'AUGUST $'
```

```
SEP DB 'SEPTEMBER $'
```

```
OCT DB 'OCTOBER $'
```

```
NOV DB 'NOVEMBER $'
```

```
DECE DB 'DECEMBER $'
```

```
MONTHTAG DW JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DECE
```

```
.code
```

```
main proc near
```

```
    mov ax,@data
```

```
    mov ds,ax
```

```
    mov es,ax
```

```
    LEA DX,TMSG ;Display Message
```

```
    MOV AH,09H
```

```
    INT 21H
```

```
    MOV AH,2CH ;Read system time
```

```
    INT 21H
```

```
    ;CH = hours, CL = minutes, DH = Seconds, DL = Hundredth of seconds
```

```
    DIGITS CH ;Display Hours
```

```
    MOV DL,':'
```

```
    MOV AH,02h
```

```
    INT 21h
```

```
    DIGITS CL ;Display Minutes
```

```
    MOV DL,':'
```

```
    MOV AH,02h
```

```
    INT 21h
```

```
    DIGITS DH ;Display Seconds
```

```
LEA DX,DMSG ;Display Message ;Showing System Date
MOV AH,09H
INT 21H
```

```
MOV AH,2AH ;Read system date
INT 21H
```

```
;DH = Month, DL = Day of Month, CX = Year, AL = Day of Week
```

```
DIGITS DL
MOV DL, ' - '
MOV AH,02h
INT 21h
```

```
;DIGITS DH
MOV AL,DH
ADD AL,AL
CBW
MOV SI,AX
LEA BX,MONTHTAG
MOV DX,[BX+SI-2]
```

```
MOV AH,09h
INT 21h
```

```
MOV DL, ' - '
MOV AH,02h
INT 21h
```

```
;YEARDISP CX
```

```
MOV DL, ', '
MOV AH,02h
INT 21h
```

```
MOV AH,2Ah
INT 21h
```

```
ADD AL,AL
CBW ;Convert byte to word
```

```
MOV SI,AX
LEA BX,DAYTAG
```

```
MOV DX, [BX+SI]
```

```
MOV AH,09h
INT 21h
```

```
MOV AH,4Ch
INT 21h
END
```

```
main endp
```

```
end main
```

## 5.2 Write a program to take input from the keyboard and print it as right justified text.

```
cur1 MACRO
    MOV AH,02
    INT 10h
    ENDM

center MACRO
    ;MOV ZX,act_len
    ;MOV kb_name[BX],07
    mov bl,act_len
    mov bh,00h
    MOV kb_name[BX+1],'$'
    MOV DL, act_len
    NEG DL
    ADD DL,79
    ;SHR DL,01
    ;NEG DL
    ;ADD DL,40    ;DL = Column
    ;MOV DH,12    ;Row Number
    MOV BH,00    ;Page No
    cur1
    ENDM

.model small
.stack 64h
.data
para1 label Byte
maxlen DB 30
act_len DB ?
kb_name DB 31 DUP (' ')
prompt DB 'Enter String : ','$' ;Prompt to the user to enter the string
.code
main proc near
    mov ax,@data
    mov ds,ax
    mov es,ax

    MOV AH,09h
    LEA DX,PROMPT
    INT 21h

    MOV AH,0Ah
    LEA DX, para1
    INT 21h

    CMP act_len,00
    JE Exit
    center
    MOV AH,09h
    LEA DX,kb_name
    INT 21h
Exit:    MOV AH,4Ch
        INT 21h
        END
```