```
       1.TITLE vowel in a string
       .MODEL SMALL
       .DATA
       STRLEN DB 80
       STRSZ DB ?
       STR DB 81 DUP('$')
       .STACK
       .CODE
       MAIN PROC FAR
       MOV AX, @DATA
       MOV DS, AX
       ;; getting the number
       LEA DX, STRLEN
       MOV AH, 0AH
       INT 21H
       ;; calculating the num of word
       MOV AX, 00
       MOV CH, 00
       MOV CL, STRSZ
       LEA BX, STR
L2:    CMP BYTE PTR[BX], 'a'
       JE ad
       CMP BYTE PTR[BX], 'e'
       JE ad
       CMP BYTE PTR[BX], 'i'
       JE ad
       CMP BYTE PTR[BX], 'o'
       JE ad
       CMP BYTE PTR[BX], 'u'
       JE ad
       CMP BYTE PTR[BX], 'A'
       JE ad
       CMP BYTE PTR[BX], 'E'
       JE ad
       CMP BYTE PTR[BX], 'I'
       JE ad
       CMP BYTE PTR[BX], 'O'
       JE ad
       CMP BYTE PTR[BX], 'U'
       JE ad
       JMP ex
ad:    INC AX
ex:    INC BX
       loop L2
```

```asm
        ;; converting to decimal
        MOV CX, 00
        MOV BX, 10
l3:     MOV DX, 00
        INC CX
        DIV BX
        CMP AX, 00
        PUSH DX
        JG l3
        ;; printing in decimal
        ;; new line character
           MOV AH, 02H
           MOV DL, 0Ah
           INT 21H
dis:    POP DX
        OR DX, 30H
        MOV AH, 02H
        INT 21H
        loop dis
        ;; ending program
        MOV AH, 4CH
        INT 21H
        MAIN ENDP

        END MAIN
```


2.   TITLE sum from 1 to n
        ;Write a program to find the sum of numbers from 1 to n. Read n from the user and display
the sum as the decimal format. (also try to display the sum in Hexadecimal format)

```asm
TITLE SUM FROM 1 TO N
.MODEL SMALL
.DATA
STRLEN DB 8
STRSZ DB ?
STR DB 9 DUP('$')
BASE DW 10
BASEH DW 10H
SUM DW ?
.STACK
.CODE

MAIN PROC FAR
MOV AX, @DATA
MOV DS, AX

;; GETTING THE NUMBER
LEA DX, STRLEN
MOV AH, 0AH
INT 21H

;; CONVERTING TO NUMBER
MOV CX, 00
MOV CL, STRSZ; strsz implicitly gets assigned with len of string in the buffer strlen
LEA DI, STR
MOV AX, 00

L1: MOV BL, [DI]
CMP BL, '0'
JB BRK
CMP BL, '9'
JA BRK

;; CONVERTING TO NUMBER
AND BL, 0FH
MUL BASE
ADD AX, BX
INC DI
LOOP L1

BRK: ;; CALCULATING THE SUM
MOV CX, AX
MOV AX, 00
```

```asm
L2: ADD AX, CX
LOOP L2
MOV SUM, AX
;; NEW LINE CHARACTER
MOV AH, 02H
MOV DL, 0AH
INT 21H


;; CONVERTING TO DECIMAL
MOV AX,SUM
MOV CX, 00
MOV BX, 10

L3: MOV DX, 00
INC CX
DIV BX
CMP AX, 00
PUSH DX
JG L3


;; PRINTING IN DECIMAL

DIS: POP DX
OR DX, 30H
MOV AH, 02H
INT 21H
LOOP DIS

;; ENDING PROGRAM
MOV AH, 4CH
INT 21H
MAIN ENDP

END MAIN
```

```
C:\>temp
5
15
C:\>temp
10
55
C:\>
```

3.   TITLE sum 10 15 bit number form memory
      .MODEL SMALL
      .DATA
      NUMS DW 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000
      SUMS DW ?
      .STACK
      .CODE
      MAIN PROC FAR
      MOV AX, @DATA
      MOV DS, AX
      ;; calculating the sum
      MOV CX, 10
      LEA BX, NUMS
      MOV AX, 00
L2:   ADD AX, [BX]
      INC BX
      INC BX
      LOOP L2
      MOV SUMS, AX
      ;; converting to decimal
      MOV CX, 00
      MOV BX, 10
l3:   MOV DX, 00
      INC CX
      DIV BX
      CMP AX, 00
      PUSH DX
      JG l3
      ;; printing in decimal
dis: POP DX
      OR DX, 30H
      MOV AH, 02H

```
        INT 21H
        loop dis
        ;; ending program
        MOV AH, 4CH
        INT 21H
        MAIN ENDP

        END MAIN
```



```
C:\>temp
5500
C:\>_
```

```
4.      TITLE count no of worlds
        .MODEL SMALL
        .DATA
        STRLEN DB 80
        STRSZ DB ?
        STR DB 81 DUP('$')
        .STACK
        .CODE
        MAIN PROC FAR
        MOV AX, @DATA
        MOV DS, AX
        ;; getting the number
        LEA DX, STRLEN
        MOV AH, 0AH
        INT 21H
        ;; calculating the num of word
        MOV AX, 01
        MOV CH, 00
        MOV CL, STRSZ
        LEA BX, STR
L2:     CMP BYTE PTR[BX], ' '
        JNE ex
        INC AX
ex:     INC BX
        loop L2
        ;; converting to decimal
        MOV CX, 00
        MOV BX, 10
l3:     MOV DX, 00
        INC CX
        DIV BX
```
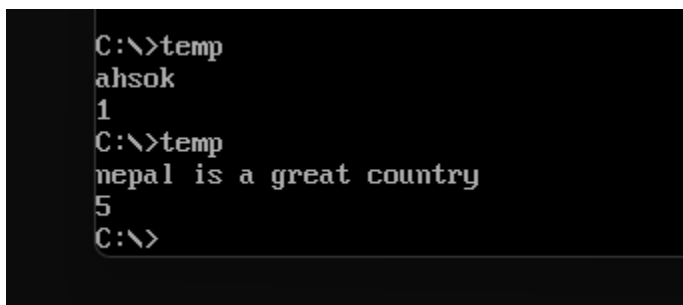
```asm
        CMP AX, 00
        PUSH DX
        JG l3
        ;; printing in decimal
        ;; new line character
            MOV AH, 02H
            MOV DL, 0Ah
            INT 21H
dis:    POP DX
        OR DX, 30H
        MOV AH, 02H
        INT 21H
        loop dis
        ;; ending program
        MOV AH, 4CH
        INT 21H
        MAIN ENDP

        END MAIN
```



```
C:\>temp
ahsok
1
C:\>temp
nepal is a great country
5
C:\>
```

```asm
5.      TITLE sum from 1 to n
        .MODEL SMALL
        .DATA
        STRLEN DB 8
        STRSZ DB ?
        STR DB 9 DUP('$')
        BASE DW 10
        BASEH DW 10h
        SUM DW ?
        .STACK
        .CODE
        MAIN PROC FAR
        MOV AX, @DATA
        MOV DS, AX
        ;; getting the number
        LEA DX, STRLEN
        MOV AH, 0AH
```

```asm
        INT 21H
        ;; converting to number
        MOV CX, 00
        MOV CL, STRSZ
        LEA DI, STR
        MOV AX, 00
l1:     MOV BL, [DI]
        CMP BL, '0'
        JB brk
        CMP BL, '9'
        JA brk
        ;; converting to number
        AND BL, 0FH
        MUL BASE
        ADD AX, BX
        INC DI
        LOOP l1
brk:    ;; calculating the sum
        MOV CX, AX
        MOV AX, 00
L2:     ADD AX, CX
        LOOP L2
        MOV SUM, AX
        ;; new line character
        MOV AH, 02H
        MOV DL, 0Ah
        INT 21H
        ;; displaying in hex
        MOV AX, SUM
        MOV CX, 4
l4:     XOR DX, DX
        MUL BASEH
        OR DL, 30h
        MOV BX, AX
        MOV AH, 02h
        INT 21H
        MOV AX, BX
        LOOP l4
        ;; converting to decimal
        MOV AX,SUM
        MOV CX, 00
        MOV BX, 10
l3:     MOV DX, 00
        INC CX
```

```
        DIV BX
        CMP AX, 00
        PUSH DX
        JG l3
        ;; new line character
        MOV AH, 02H
        MOV DL, 0Ah
        INT 21H
        ;; printing in decimal
dis:    POP DX
        OR DX, 30H
        MOV AH, 02H
        INT 21H
        loop dis
        ;; ending program
        MOV AH, 4CH
        INT 21H
        MAIN ENDP

        END MAIN
```
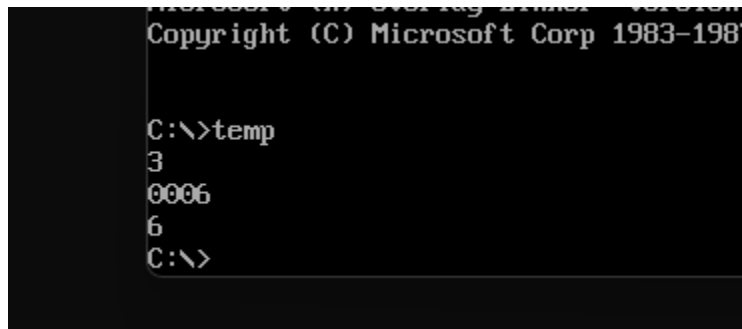

DISCUSSION
In this lab, we learned the advance programs of assembly language. We learned to take ascii codes from keyboard and do operations in the inputs. We learned to display numbers in decimal and hexadecimal. We also learned write programs with multiple or like syntax to check vowel letters. Taking multiple digit numbers from user and displaying was also taught in this lab.

CONCLUSION
Hence, we learned to write useful programs in assembly taking user input, processing the input and displaying the output.