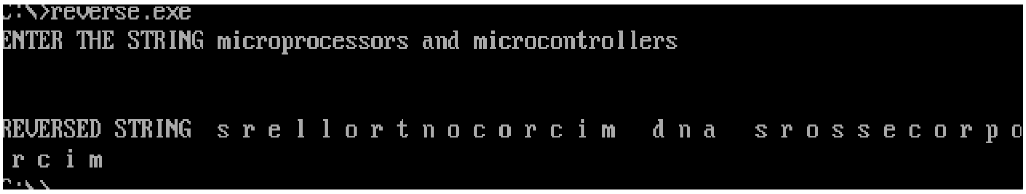
Name : BHAGYA A JAI
ROLL NO : B21CSB18
EXPERIMENT : STRING REVERSAL

PROGRAM

```
DATA SEGMENT
ST1 DB 100 DUP(?)
MSG1 DB "ENTER THE STRING $"
MSG2 DB "REVERSED STRING $"
DATA ENDS

CODE SEGMENT
ASSUME CS:CODE
START:
MOV AX,DATA
MOV DS,AX
LEA DX,MSG1
MOV AH,09H
INT 21H
LEA SI,ST1
MOV CL,00H
L1:
MOV AH,01H
INT 21H
CMP AL,0DH
JZ L2
INC CL
INC SI
MOV [SI],AL
INC SI
INC CL
JMP L1
L2:
MOV DL,0AH
INT 21H
MOV DL,0DH
INT 21H
LEA DX,MSG2
MOV AH,09H
INT 21H
L3:
MOV DL,[SI]
MOV AH,02H
INT 21H
DEC SI
DEC CL
CMP CL,00H
JNZ L3
MOV AH,4CH
INT 21H
CODE ENDS
END START
```

OUTPUT



```
C:\>reverse.exe
ENTER THE STRING microprocessors and microcontrollers


REVERSED STRING  s r e l l o r t n o c o r c i m   d n a   s r o s s e c o r p o
 r c i m
C:\>
```

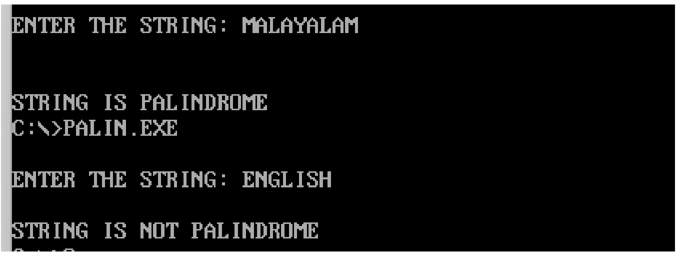
Name : BHAGYA A JAI
ROLL NO : B21CSB18
EXPERIMENT : PALINDROME CHECKING

PROGRAM

```
DATA SEGMENT
MSG1 DB 10,13, "ENTER THE STRING: $"
MSG2 DB 10,13, "STRING IS PALINDROME $"
MSG3 DB 10,13, "STRING IS NOT PALINDROME $"
NEW Dw 10,13, "$"
INST DB 20 DUP(0)
DATA ENDS

CODE SEGMENT
ASSUME CS:CODE,DS: DATA
START:
MOV AX, DATA
MOV DS,AX
LEA DX, MSG1
MOV AH, 09H
INT 21H
MOV BX,00

UP:
MOV AH,01H
INT 21H
CMP AL, 0DH
JE DOWN
MOV [INST+BX], AL
INC BX
LOOP UP

DOWN:
MOV DI,0
DEC BX
JNZ CHECK

CHECK:
MOV AL, [INST+BX]
CMP AL, [INST+DI]
JNE FAIL
INC DI
DEC BX
JNZ CHECK
LEA DX, NEW
MOV AH,09h
INT 21H
LEA DX,MSG2
MOV AH, 09H
INT 21H
JMP FINISH

FAIL:
LEA DX, MSG3
```

```
MOV AH, 09H
INT 21H

FINISH:
MOV AH,4CH
INT 21h
CODE ENDS
END START
```

OUTPUT



```
ENTER THE STRING: MALAYALAM


STRING IS PALINDROME
C:\>PALIN.EXE

ENTER THE STRING: ENGLISH

STRING IS NOT PALINDROME
```

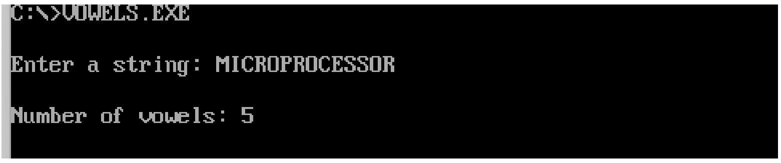
Name : BHAGYA A JAI
ROLL NO : B21CSB18
EXPERIMENT : COUNTING VOWELS

PROGRAM

```
Display macro msg
      MOV AH,09H
      MOV DX,offset msg
      INT 21H
ENDM
.MODEL small
.stack 100H
.Data
      CR equ 0DH
      LF equ 0AH
      msg1 DB 'Enter a string: ','$'
      msg2 DB CR,LF,'Number of vowels: ','$'
      str DB 20 dup(0)
      vow DB 03 dup(0)
.Code
      START:
            MOV AX,@Data
            MOV DS,AX
                  MOV SI,offset str
                  Display msg1
                  MOV CX,00H
                  MOV BX,00H
      READ:
            MOV AH,01H
                  INT 21H
                  CMP AL,CR
                  JE AHEAD
                  MOV [SI],AL
                  INC SI
                  INC CX
                  JMP READ
      AHEAD:
            MOV SI,offset str
      CHECK:
            MOV AL,[SI]
            CMP AL,'a'
            JE LOOP2
            CMP AL,'e'
            JE LOOP2
            CMP AL,'i'
            JE LOOP2
            CMP AL,'o'
            JE LOOP2
            CMP AL,'u'
            JE LOOP2
      LOOP1:
            INC SI
            LOOP CHECK
            JMP NEXT
      LOOP2:
            INC BX
```

```
           JMP LOOP1
      NEXT:
           MOV AX,BX
           MOV SI,offset vow
           CALL hextoasc
           DISPLAY msg2
           DISPLAY vow
           MOV AH,4CH
           INT 21H
      hextoasc PROC NEAR
           MOV CX,00H
                 MOV BX,0AH
           REP1:
                 MOV DX,00H
                 DIV BX
                 ADD DL,'0'
                 PUSH DX
                 INC CX
                 CMP AX,0AH
                 JGE REP1
                 ADD AL,'0'
                 MOV [SI],AL
           REP2:
                 POP AX
                 INC SI
                 MOV [SI],AL
                 LOOP REP2
                 INC SI
                 MOV AL,'$'
                 MOV [SI],AL
                 RET
        hextoasc ENDP
END START
```

OUTPUT



```
C:\>VOWELS.EXE

Enter a string: MICROPROCESSOR

Number of vowels: 5
```

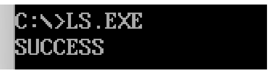
Name : BHAGYA A JAI
ROLL NO : B21CSB18
EXPERIMENT : LINEAR SEARCH

PROGRAM

```
DATA SEGMENT
 ARR DB 10,20,30,40
 COUNT EQU 5
 ITEM EQU 20
 MSG1 DB "SUCCESS",13,10,'$'
 MSG2 DB "FAILED",13,10,'$'
DATA ENDS
CODE SEGMENT
ASSUME CS:CODE,DS:DATA
START:
 MOV AX,DATA
 MOV DS,AX
 MOV SI,OFFSET ARR
 MOV CL,COUNT
 MOV AL,ITEM
NEXT:
CMP AL,[SI]
 JE FOUND
 INC SI
 LOOP NEXT
 MOV AH,09H
 MOV DX,OFFSET MSG2
 INT 21H
 JMP EXIT
FOUND:
MOV AH,09H
 MOV DX,OFFSET MSG1
 INT 21H
EXIT:
MOV AH,4CH
 INT 21H
CODE ENDS
END START
```

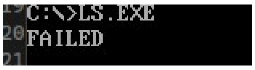
OUTPUT

item set as 20:



```
C:\>LS.EXE
SUCCESS
```

item set as 15:



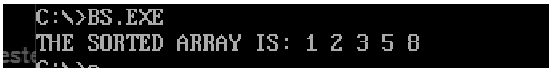
```
C:\>LS.EXE
FAILED
```

Name : BHAGYA A JAI
ROLL NO : B21CSB18
EXPERIMENT : BUBBLE SORT

PROGRAM

```
.MODEL SMALL
.STACK
.Data
LIST DB 03H,08H,02H,01H,05H
N DW $-LIST
msg DB 'THE SORTED ARRAY IS: $'
.Code
MOV AX,@Data
MOV DS,AX
MOV BX,N
DEC BX
NXTPASS:
MOV CX,BX
MOV SI,00H
NXTCOMP:
MOV AL,LIST[SI]
INC SI
CMP AL,LIST[SI]
JB NEXT ; ascending order, if AL>LIST[SI]
XCHG AL,LIST[SI]
MOV LIST[SI-1],AL
NEXT:
LOOP NXTCOMP
DEC BX
JNZ NXTPASS
LEA DX,MSG
MOV AH,09H
INT 21H
MOV BX,N
MOV SI,00H
AGAIN:
MOV AL,LIST[SI]
ADD AL,'0'
MOV DL,AL
MOV AH,02H
INT 21H
MOV AH,02H
MOV DL,' '
INT 21H
INC SI
DEC BX
JNZ AGAIN
MOV AH,4CH
INT 21H
END
```

OUTPUT



```
C:\>BS.EXE
THE SORTED ARRAY IS: 1 2 3 5 8
C:\>
```