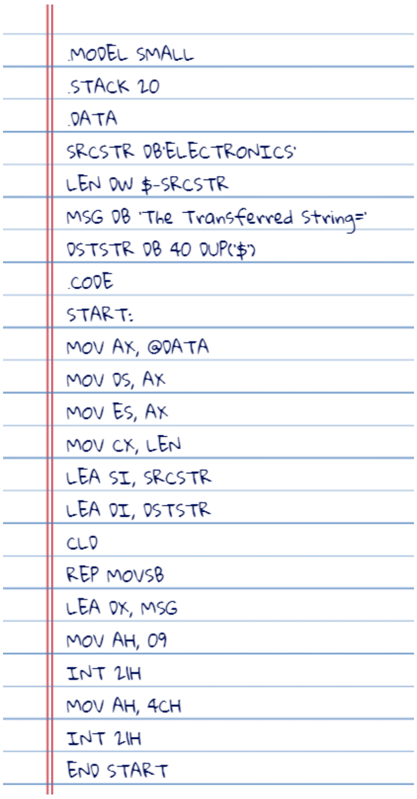
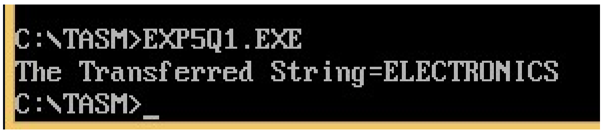
P1:

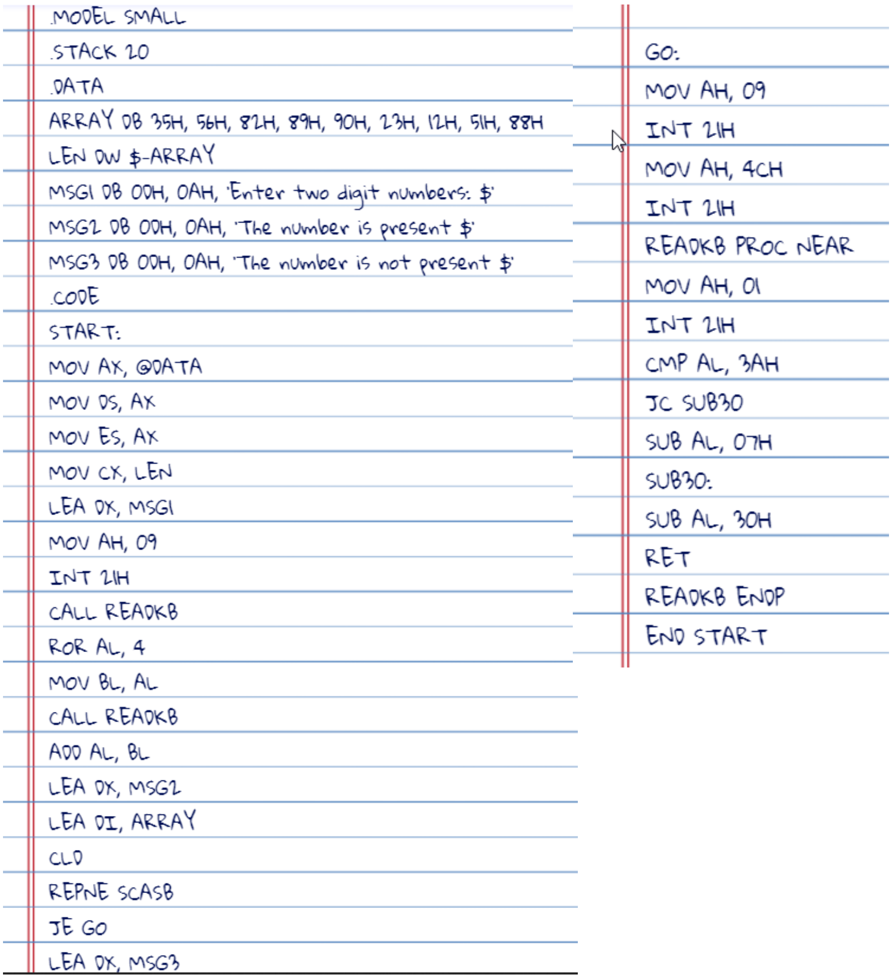


```
.MODEL SMALL
.STACK 20
.DATA
SRCSTR DB'ELECTRONICS'
LEN DW $-SRCSTR
MSG DB 'The Transferred String='
DSTSTR DB 40 DUP('$')
.CODE
START:
MOV AX, @DATA
MOV DS, AX
MOV ES, AX
MOV CX, LEN
LEA SI, SRCSTR
LEA DI, DSTSTR
CLD
REP MOVSB
LEA DX, MSG
MOV AH, 09
INT 21H
MOV AH, 4CH
INT 21H
END START
```

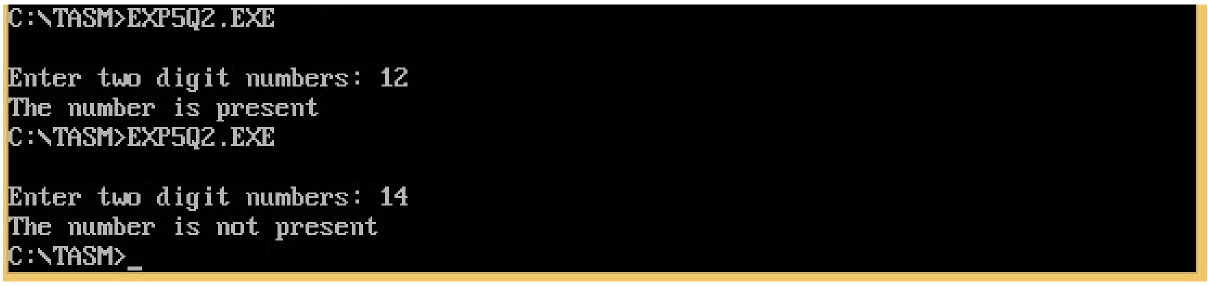


```
C:\TASM>EXP5Q1.EXE
The Transferred String=ELECTRONICS
C:\TASM>_
```

P2:

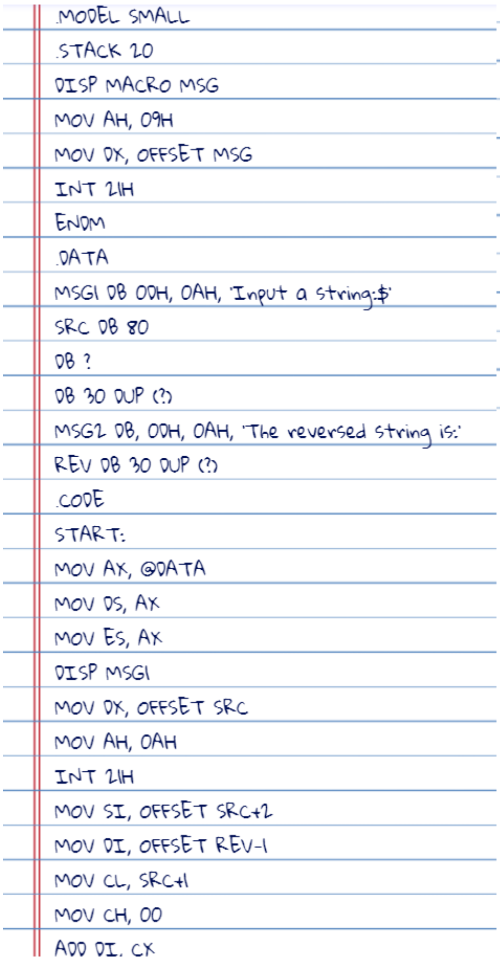


```
.MODEL SMALL
.STACK 20
.DATA
ARRAY DB 35H, 56H, 82H, 89H, 90H, 23H, 12H, 51H, 88H
LEN DW $-ARRAY
MSG1 DB 0DH, 0AH, 'Enter two digit numbers: $'
MSG2 DB 0DH, 0AH, 'The number is present $'
MSG3 DB 0DH, 0AH, 'The number is not present $'
.CODE
START:
MOV AX, @DATA
MOV DS, AX
MOV ES, AX
MOV CX, LEN
LEA DX, MSG1
MOV AH, 09
INT 21H
CALL READKB
ROR AL, 4
MOV BL, AL
CALL READKB
ADD AL, BL
LEA DX, MSG2
LEA DI, ARRAY
CLD
REPNE SCASB
JE GO
LEA DX, MSG3
GO:
MOV AH, 09
INT 21H
MOV AH, 4CH
INT 21H
READKB PROC NEAR
MOV AH, 01
INT 21H
CMP AL, 3AH
JC SUB30
SUB AL, 07H
SUB30:
SUB AL, 30H
RET
READKB ENDP
END START
```

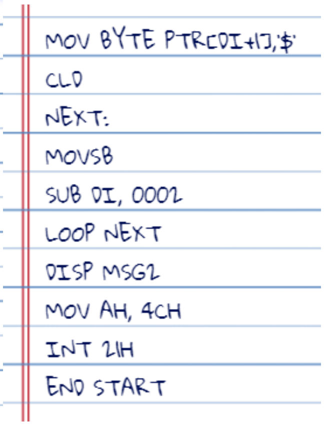


```
C:\TASM>EXP5Q2.EXE

Enter two digit numbers: 12
The number is present
C:\TASM>EXP5Q2.EXE

Enter two digit numbers: 14
The number is not present
C:\TASM>_
```

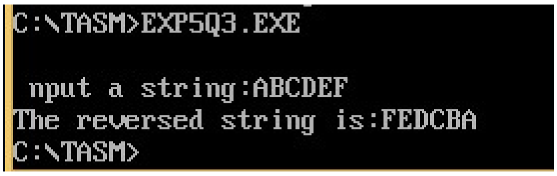
P3:



```
.MODEL SMALL
.STACK 20
DISP MACRO MSG
MOV AH, 09H
MOV DX, OFFSET MSG
INT 21H
ENDM
.DATA
MSG1 DB 0DH, 0AH, 'Input a string:$'
SRC DB 80
DB ?
DB 30 DUP (?)
MSG2 DB, 0DH, 0AH, 'The reversed string is:'
REV DB 30 DUP (?)
.CODE
START:
MOV AX, @DATA
MOV DS, AX
MOV ES, AX
DISP MSG1
MOV DX, OFFSET SRC
MOV AH, 0AH
INT 21H
MOV SI, OFFSET SRC+2
MOV DI, OFFSET REV-1
MOV CL, SRC+1
MOV CH, 00
ADD DI, CX
```

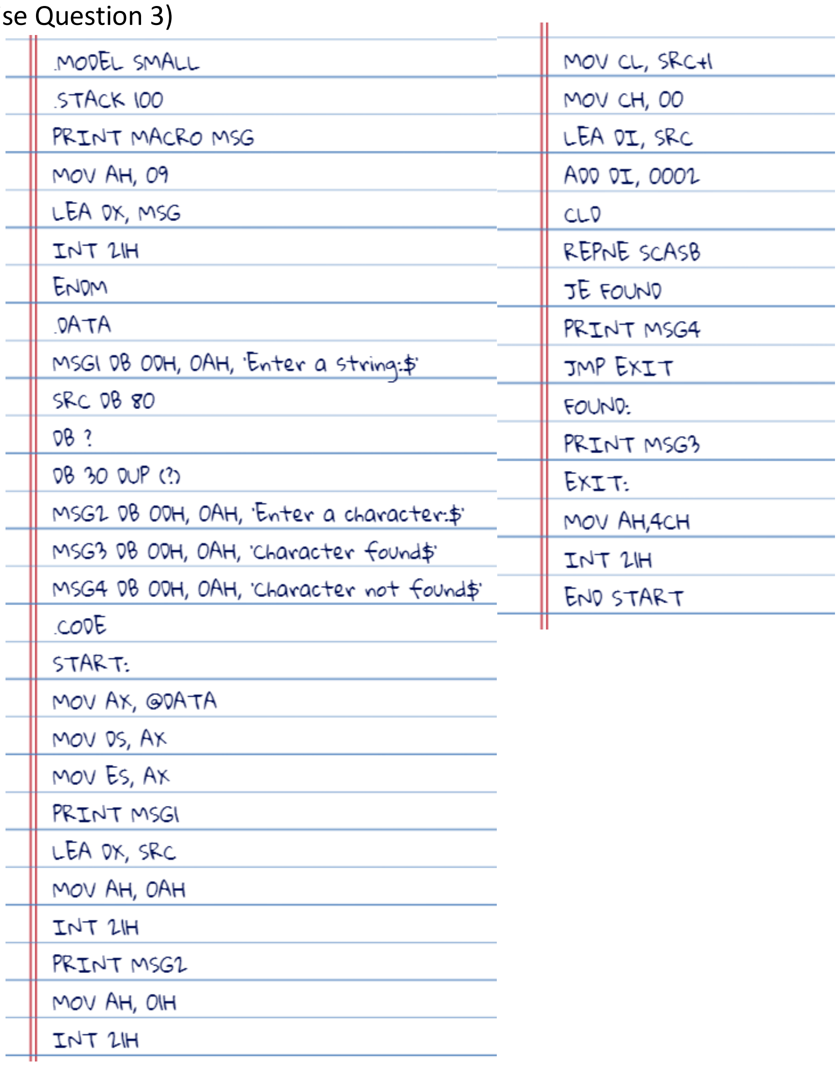


```
MOV BYTE PTR[DI+1],'$'
CLD
NEXT:
MOVSB
SUB DI, 0002
LOOP NEXT
DISP MSG2
MOV AH, 4CH
INT 21H
END START
```

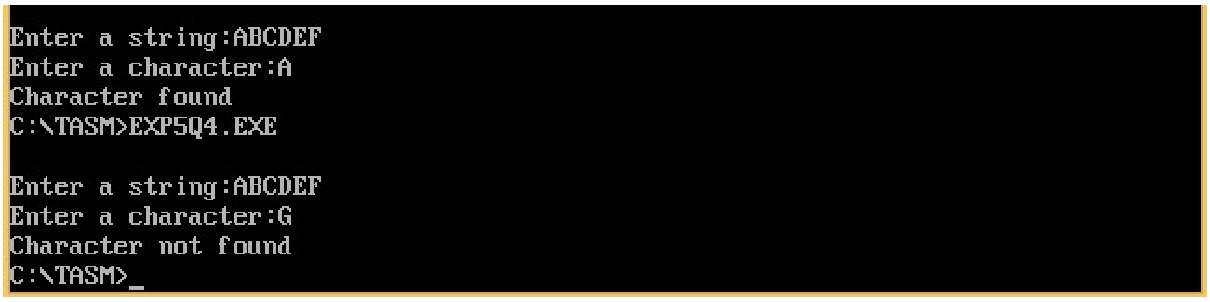


```
C:\TASM>EXP5Q3.EXE

 nput a string:ABCDEF
The reversed string is:FEDCBA
C:\TASM>
```

P4: (Exercise Question 3)



```
.MODEL SMALL
.STACK 100
PRINT MACRO MSG
MOV AH, 09
LEA DX, MSG
INT 21H
ENDM
.DATA
MSG1 DB 0DH, 0AH, 'Enter a string:$'
SRC DB 80
DB ?
DB 30 DUP (?)
MSG2 DB 0DH, 0AH, 'Enter a character:$'
MSG3 DB 0DH, 0AH, 'Character found$'
MSG4 DB 0DH, 0AH, 'Character not found$'
.CODE
START:
MOV AX, @DATA
MOV DS, AX
MOV ES, AX
PRINT MSG1
LEA DX, SRC
MOV AH, 0AH
INT 21H
PRINT MSG2
MOV AH, 01H
INT 21H
MOV CL, SRC+1
MOV CH, 00
LEA DI, SRC
ADD DI, 0002
CLD
REPNE SCASB
JE FOUND
PRINT MSG4
JMP EXIT
FOUND:
PRINT MSG3
EXIT:
MOV AH,4CH
INT 21H
END START
```



```
Enter a string:ABCDEF
Enter a character:A
Character found
C:\TASM>EXP5Q4.EXE

Enter a string:ABCDEF
Enter a character:G
Character not found
C:\TASM>_
```