NAME: **GINI CHACKO**

SEMESTER: **IV**

CLASS: **SE COMPS B**

BATCH: **B**

ROLL: **8942**

TOPIC: **MP EXPERIMENT 6:**

A.] Block transfer from source to destination

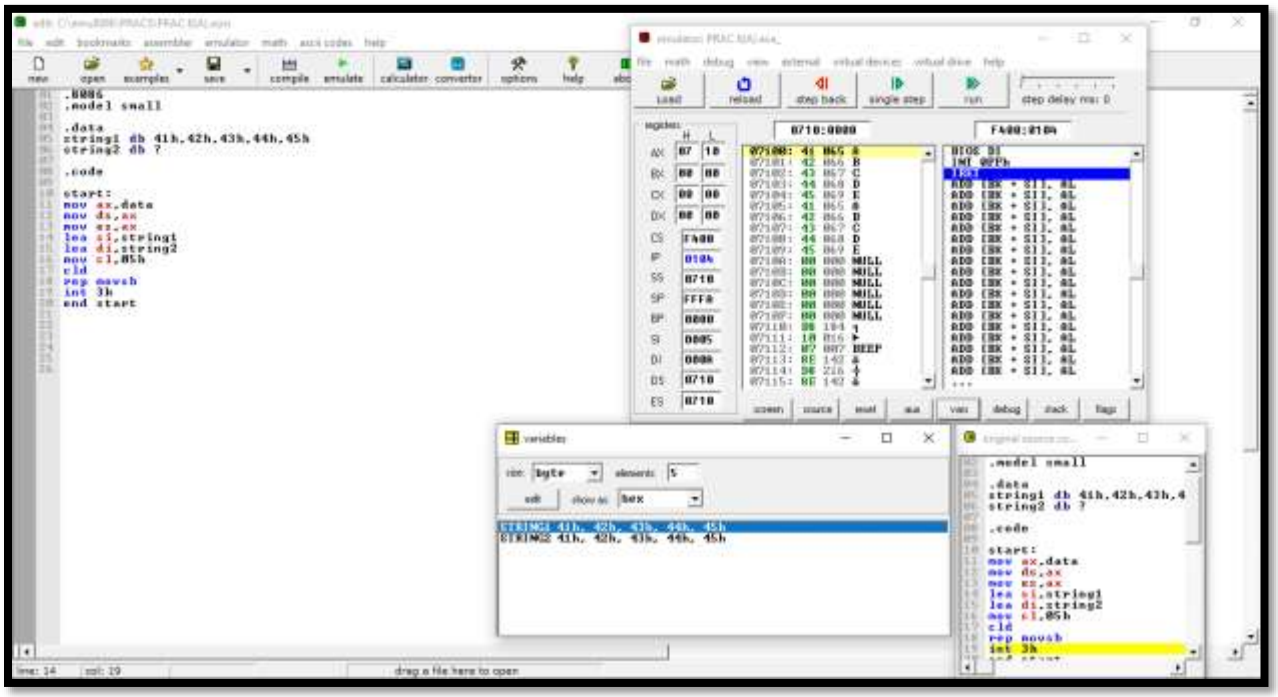
B.] Check whether it is palindrome or not

A.] Block transfer from source to destination

CODE:

```
.8086
.model small

.data
string1 db 41h,42h,43h,44h,45h
string2 db ?

.code

start:
mov ax,data
mov ds,ax
mov es,ax
lea si,string1
lea di,string2
mov cl,05h
cld
rep movsb
int 3h
end start
```

OUTPUT:



```
.8086
.model small

.data
string1 db 41h,42h,43h,44h,45h
string2 db ?

.code

start:
mov ax,data
mov ds,ax
mov es,ax
lea si,string1
lea di,string2
mov cl,05h
cld
rep movsb
int 3h
end start
```

```
AX 07 10
BX 00 00
CX 00 00
DX 00 00
CS F400
IP 0104
SS 0710
SP FFFE
BP 0000
SI 0005
DI 000A
DS 0710
ES 0710
```

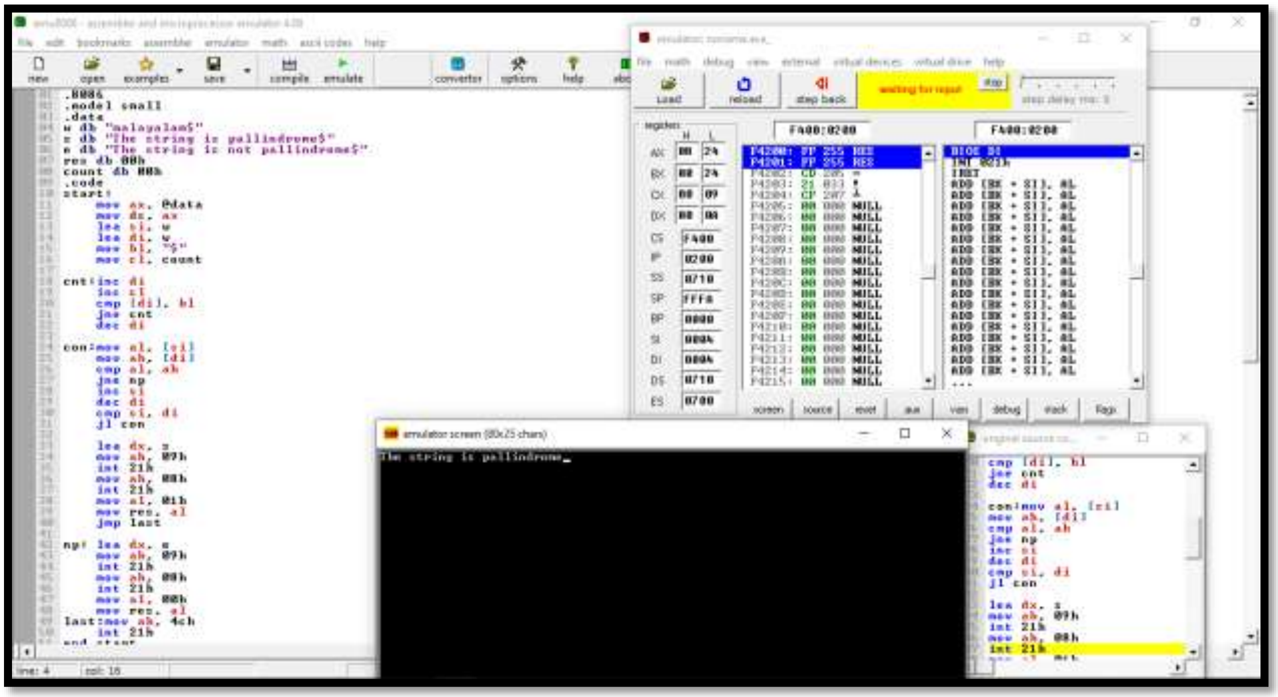
```
STRING1 41h, 42h, 43h, 44h, 45h
STRING2 41h, 42h, 43h, 44h, 45h
```

B.] Check whether it is palindrome or not

CODE:

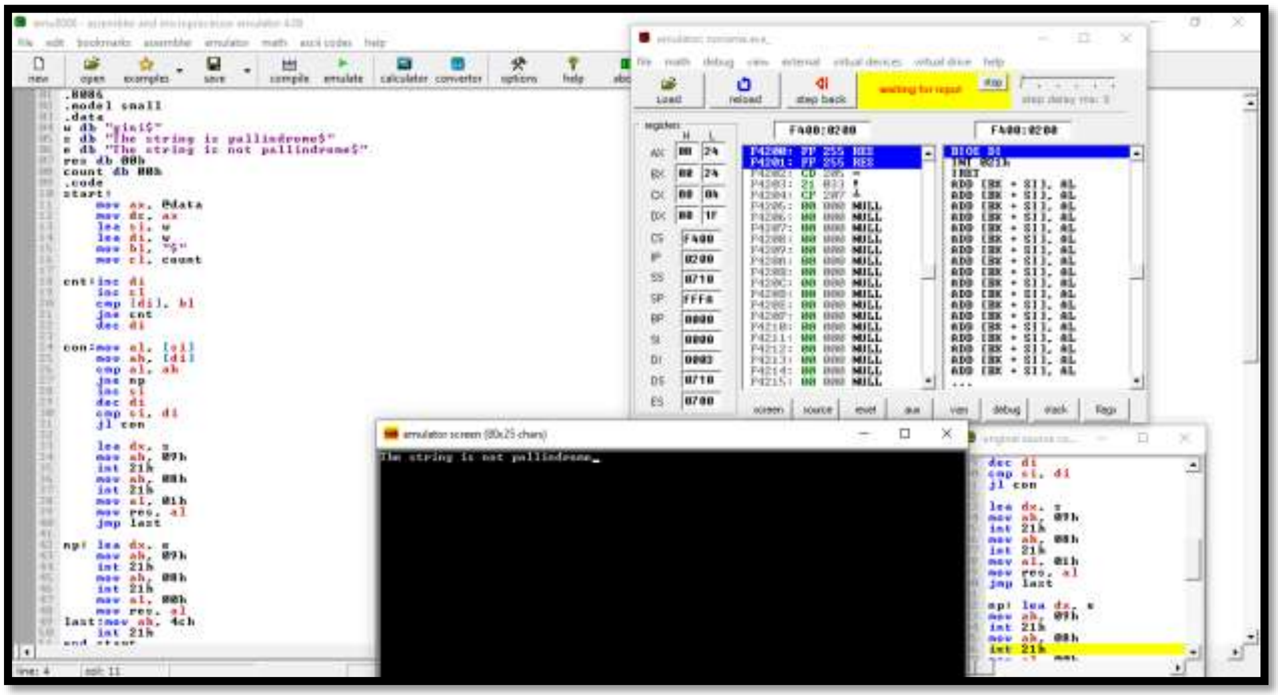
```
.8086
.model small
.data
w db "gini$"
s db "The string is pallindrome$"
e db "The string is not pallindrome$"
res db 00h
count db 00h
.code
start:
    mov ax, @data
    mov ds, ax
    lea si, w
    lea di, w
    mov bl, "$"
    mov cl, count

cnt:inc di
    inc cl
    cmp [di], bl
    jne cnt
    dec di

con:mov al, [si]
    mov ah, [di]
    cmp al, ah
    jne np
    inc si
    dec di
    cmp si, di
    jl con

    lea dx, s
    mov ah, 09h
    int 21h
    mov ah, 08h
    int 21h
    mov al, 01h
    mov res, al
    jmp last

np: lea dx, e
    mov ah, 09h
    int 21h
    mov ah, 08h
    int 21h
    mov al, 00h
    mov res, al
last:mov ah, 4ch
    int 21h
end start
```

OUTPUT:



```
.8086
.model small
.data
w db "malayalam$"
s db "The string is pallindrome$"
e db "The string is not pallindrome$"
res db 00h
count db 00h
.code
start:
     mov ax, @data
     mov ds, ax
     lea si, w
     lea di, w
     mov bl, "$"
     mov cl, count

cnt:inc di
     inc cl
     cmp [di], bl
     jne cnt
     dec di

con:mov al, [si]
     mov ah, [di]
     cmp al, ah
     jne np
     inc si
     dec di
     cmp si, di
     jl con

     lea dx, s
     mov ah, 09h
     int 21h
     mov ah, 08h
     int 21h
     mov al, 01h
     mov res, al
     jmp last

np: lea dx, e
     mov ah, 09h
     int 21h
     mov ah, 08h
     int 21h
     mov al, 00h
     mov res, al
last:mov ah, 4ch
     int 21h
end start
```

```
The string is pallindrome_
```



```
.8086
.model small
.data
w db "gini$"
s db "The string is pallindrome$"
e db "The string is not pallindrome$"
res db 00h
count db 00h
.code
start:
     mov ax, @data
     mov ds, ax
     lea si, w
     lea di, w
     mov bl, "$"
     mov cl, count

cnt:inc di
     inc cl
     cmp [di], bl
     jne cnt
     dec di

con:mov al, [si]
     mov ah, [di]
     cmp al, ah
     jne np
     inc si
     dec di
     cmp si, di
     jl con

     lea dx, s
     mov ah, 09h
     int 21h
     mov ah, 08h
     int 21h
     mov al, 01h
     mov res, al
     jmp last

np: lea dx, e
     mov ah, 09h
     int 21h
     mov ah, 08h
     int 21h
     mov al, 00h
     mov res, al
last:mov ah, 4ch
     int 21h
end start
```

```
The string is not pallindrome_
```

GINI CHACKO 8942

POSTLAB QUESTIONS:

1. **Explain any 5 string instructions with examples.**

Ans:

String is a group of bytes/words and their memory is always allocated in a sequential order. String is either referred as byte string or word string.

OPCODE	OPERAND	EXPLANATION	EXAMPLE
REP	instruction	repeat the given instruction till CX != 0	REP MOVSB
REPE	instruction	repeat the given instruction while CX = 0	REPE
REPZ	instruction	repeat the given instruction while ZF = 1	REPZ
REPNE	instruction	repeat the given instruction while CX != 0	REPNE
REPNZ	instruction	repeat the given instruction while ZF = 0	REPNZ
MOVSB	none	moves contents of byte given by DS:SI into ES:DI	MOVSB
MOVSW	none	moves contents of word given by DS:SI into ES:DI	MOVSW
MOVD	none	moves contents of double word given by DS:SI into ES:DI	MOVD