

Expt. No. \_\_\_\_\_

IBM19CS179

Page No. \_\_\_\_\_

• model small

• data

str1 db 10 dup(0)

str2 db 10 dup(0)

len1 db 00

len2 db 00

msg1 db 0ah, 0ah, "enter first string \$"

msg2 db 0ah, 0ah, "enter second string \$"

msg3 db 0ah, 0ah, "strings are equal \$"

msg4 db 0ah, 0ah, "strings are not equal \$"

msg5 db 0ah, 0ah, "length of first string is \$"

msg6 db 0ah, 0ah, "length of second string is \$"

msg7 db 0ah, 0ah, "length of string is \$"

• code

mov ax, @data

mov ds, ax

lea dx, msg1

mov ds, ax

int 21h

mov si, 00

back1: mov ah, 01h

int 21h

cmp al, 0ah

Teacher's Signature : \_\_\_\_\_

```
je next1  
mov str1[si], al  
inc si  
inc len1  
jmp back1
```

```
next1: lea dx, msg2  
mov ah, 09h  
int 21h  
mov si, 00
```

```
back2: mov ah, 01h  
int 21h  
cmp al, 0dh  
je next2  
mov str2[si], al  
inc si  
inc len2  
jmp back2
```

```
next2: mov al, len1  
cmp al, len2  
jne notequal
```

; when length of both strings are equal that  
is len1 = len2

```
mov si, 00
```

```
mov di, 00
```

```
mov cx, len1 ; mov cx, len2
```



back 3 : mov al, str 1 [si]  
cmp al, str 2 [di]  
jne not equal  
inc si  
dec cl  
jnz back 3 ; can use loop statements

lea dx, msg 3  
mov ah, 09h  
int 21h

lea dx, msg 7  
mov ah, 09h  
int 21h

mov dl, len1 ; mov dl, len2  
add dl, 30h  
mov ah, 02h  
int 21h  
jmp last

not equal : lea dx, msg 4  
mov ah, 09h  
int 21h

lea dx, msg 5  
mov ah, 09h  
int 21h

```
mov dl, len1  
add dl, 30h  
mov ah, 02h  
int 21h
```

```
lea dx, msg6  
mov ah, 09h  
int 21h  
mov dl, len2  
add dl, 30h  
mov ah, 02h  
int 21h
```

```
last : mov ah, 4ch  
int 21h
```

```
end.
```

Teacher's Signature : \_\_\_\_\_