horizontal line

**Assignment 06 | MFP**

**CE-092**

Assignment submission for Microprocessor Fundamentals and Programming subject week 6.

nevilparmar24@gmail.com

**─**

**Task 1:**

Write a program to concatenate two strings STR1 and STR2 and store the result in the string STR3.

Code:

data segment

str1 *db* 20 dup(0)

str2 *db* 20 dup(0)

str3 *db* 40 dup(0)

msg1 *db* 10,13,'enter the first string: $'

msg2 *db* 10,13,'enter the second string: $'

msg3 *db* 10,13,'concatenated string is: $'

new *db* 10,13,'$'

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,000

up1:

mov ah,01h

int 21h

cmp al,0dh

je down1

mov [str1+bx],al

inc bx

jmp up1

down1:

lea dx,new

mov ah,09h

int 21h

lea dx,msg2

mov ah,09h

int 21h

mov cx,bx

mov bx,00

up2:

mov ah,01h

int 21h

cmp al,0dh

je down2

mov [str2+bx],al

inc bx

jmp up2

down2:

push bx

mov di,0

mov si,0

up3:

mov al,[str1+di]

mov [str3+si],al

inc si

inc di

loop up3

pop cx

mov di,0

up4:

mov al,[str2+di]

mov [str3+si],al

inc si

inc di

loop up4

lea dx,new

mov ah,09h

int 21h

lea dx,msg3

mov ah,09h

int 21h

mov [str3+si],'$'

lea dx,str3

mov ah,09h

int 21h

mov ah,4ch

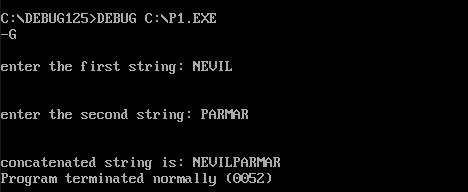
int 21h

code ends

end start

end

Output:



**Task 2:**

Write an assembly language to count the number of occurrences of a substring in a given string.

Code:

data segment

space *db* 0DH,0AH,"$"

string *db* "aaaabbcabcabc"

substring *db* "abc$"

result *db* 0DH,0AH,"Number of occurrence is :- $"

count *dw* 00h

data ends

print macro msg

mov dx, offset msg

mov ah, 09h

int 21h

endm

code segment

assume ds:data, cs:code, es:data

start:

*cld*

mov ax, data

mov ds, ax

mov es, ax

mov bx, 00h

lea di, string

mov ax, di

mov dx, di

add dx, 1Fh

sub dx, 03h

inc dx

up:

mov di, ax

lea si, substring

mov cx, 03h

repe cmpsb

jz next

prev:

add ax, 01h

cmp ax, dx

jnz up

jmp exit

next:

inc bx

mov cx, 03h

jmp prev

exit:

print string

print space

print substring

print space

print result

mov [count], bx

mov dl, *byte* ptr count[0]

add dl, 30h

mov ah, 02h

int 21h

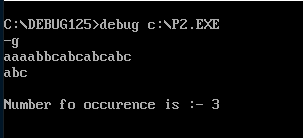
mov ah,4CH

int 21h

code ends

end start

Output:



**Task 3:**

In a string “Good Morning”, write a program to replace the substring “Morning” with “Evening” and display “Good Evening”.

Code:

data segment

new\_greet *db* "Evening$"

greeting *db* "Good Morning$"

data ends

print macro msg

mov dx,offset msg

mov ah,09h

int 21h

endm

code segment

assume ds:data,cs:code

start:

*cld*

mov ax, data

mov ds, ax

mov es, ax

lea si, new\_greet

lea di, greeting

mov al, 'M'

mov cx, 0ch

repne scasb

dec di

rep movsb

print greeting

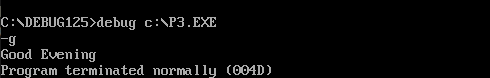
mov ah,4CH

int 21h

code ends

end start

Output:



**Task 4:**

Program to enter two strings, Find the characters that match in both the strings, store these characters.

Code:

data segment

input\_msg1 *db* 0Dh, 0Ah, "Enter first String :- $"

input\_msg2 *db* 0Dh, 0Ah, "Enter second String :- $"

output\_msg *db* 0Dh, 0Ah, "Common characters are :-$"

heading *db* 0DH, 0AH, "Program to store common characters : $"

string1 *db* 100 dup('$')

string2 *db* 100 dup('$')

space *db* 0DH, 0AH, "$"

data ends

print\_msg macro msg

mov dx, offset msg

mov ah, 09h

int 21h

endm

code segment

assume ds:data, cs:code

start:

*cld*

mov ax, data

mov ds, ax

mov es, ax

print\_msg heading

print\_msg input\_msg1

mov dx, offset string1 *; Input string1*

mov ah, 0ah

int 21h

mov cl, string1 + 1

mov ch, 00h

lea si,string1 + 2

print\_msg space

print\_msg input\_msg2

mov dx,offset string2 *; Input string2*

mov ah,0ah

int 21h

lea di,string2 + 2

print\_msg space

print\_msg output\_msg

mov ax, 0000h

mov bx, cx

again:

mov cl, string1 + 1

mov ch, 00h

lea di, string2 + 2

lodsb

repne scasb

jz char\_disp

up:

dec bx

cmp bx,0000h

jne again

jmp Exit

char\_disp:

mov dl, al

mov ah, 02h

int 21h

jmp up

Exit:

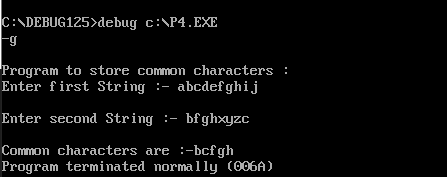
mov ah,4CH

int 21h

code ends

end start

Output:



Nevil Parmar

CE-092

https://nevilparmar.me