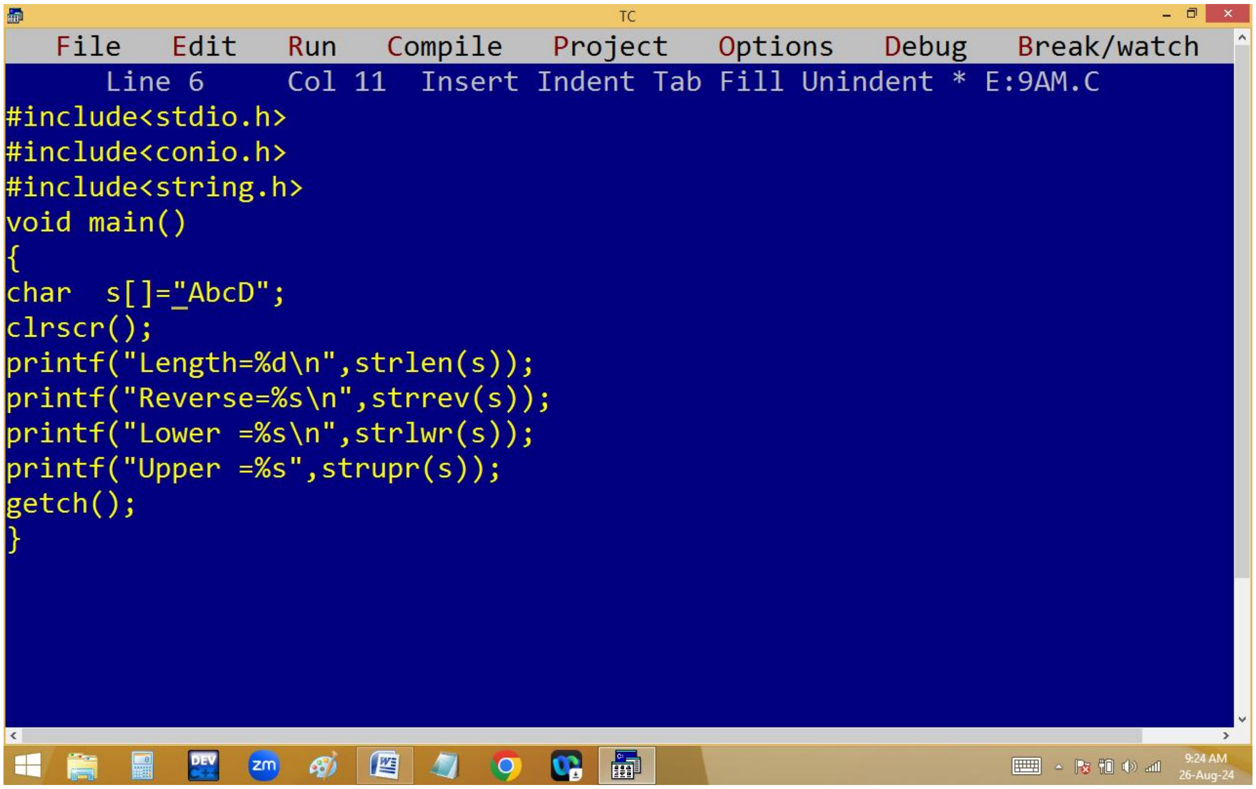
STRING LIBRARY FUNCTIONS

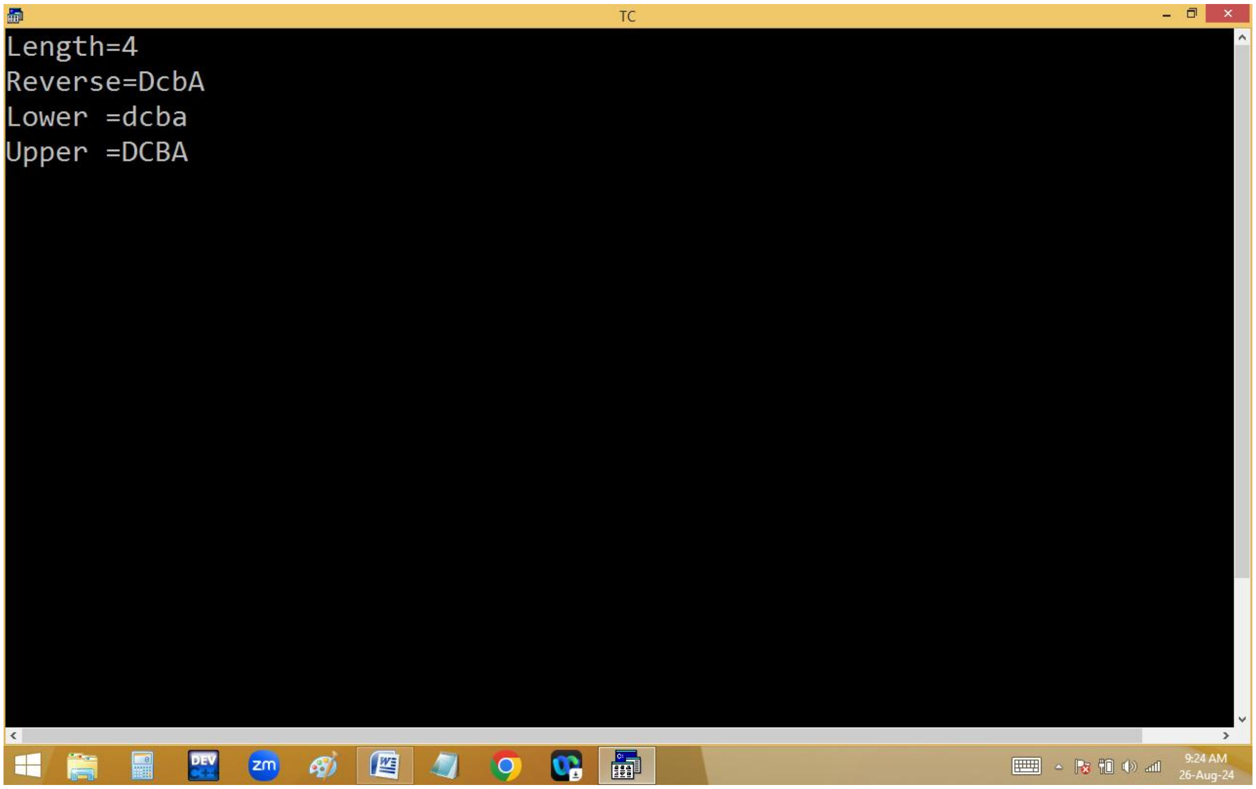
To manage string operations c language provides some of the predefined functions available in <string.h>

They are

1. **strlen(): It return string length.**
   Syntax: strlen(string);
2. **strrev(): it return reverse string.**
   Syntax: strrev(string);
3. **strlwr(): converts string into lower case.**
   Syntax: strlwr(string);
4. **strupr(): converts string into upper case.**
   Syntax: strupr(string);



```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char  s[]="AbcD";
clrscr();
printf("Length=%d\n",strlen(s));
printf("Reverse=%s\n",strrev(s));
printf("Lower =%s\n",strlwr(s));
printf("Upper =%s",strupr(s));
getch();
}
```



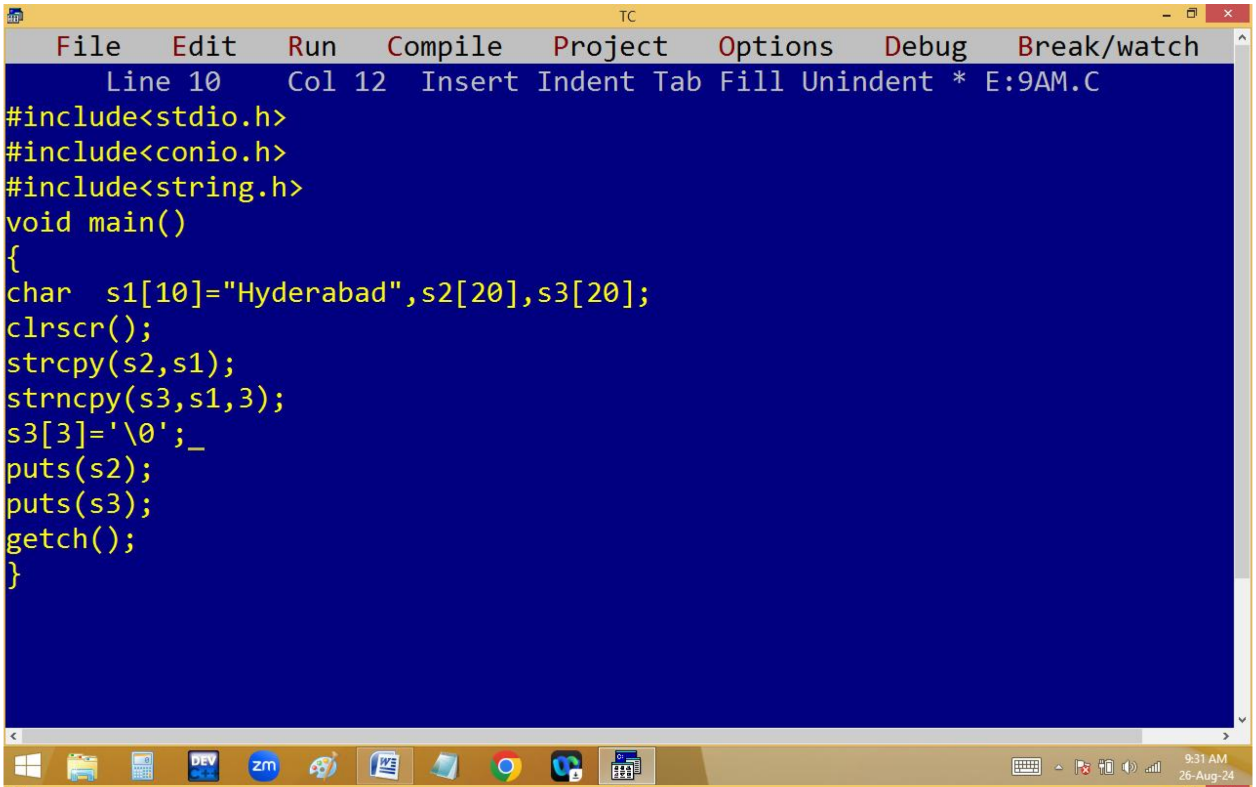
```
Length=4
Reverse=DcbA
Lower =dcba
Upper =DCBA
```

5. **strcpy(): It copies the source string into destination string.**

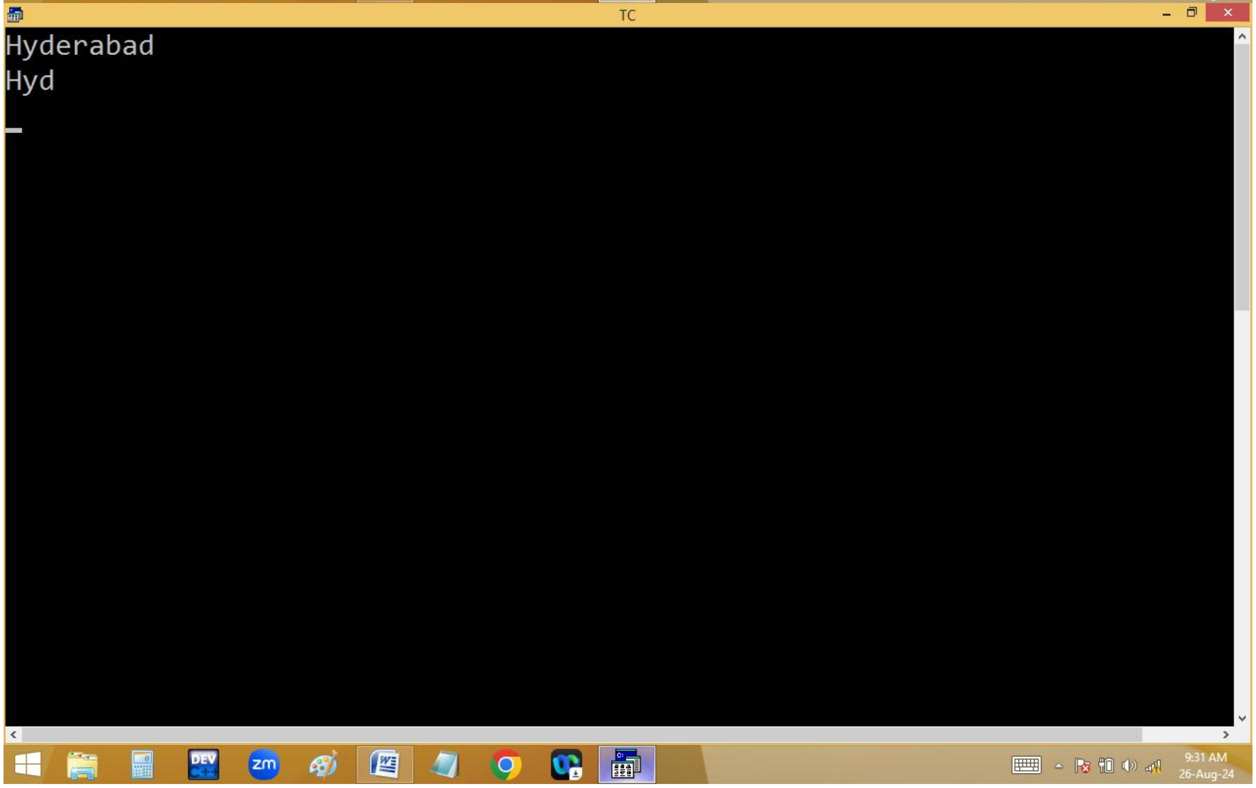
   Syntax: strcpy(destination string, source string);

6. **strncpy(): It copies specified no of char into destination string.**

   Syntax:strncpy(dest string, source string, no of char);


TC
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 12 Insert Indent Tab Fill Unindent * E:9AM.C
9:31 AM
26-Aug-24

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char  s1[10]="Hyderabad",s2[20],s3[20];
clrscr();
strcpy(s2,s1);
strncpy(s3,s1,3);
s3[3]='\0';_
puts(s2);
puts(s3);
getch();
}
```



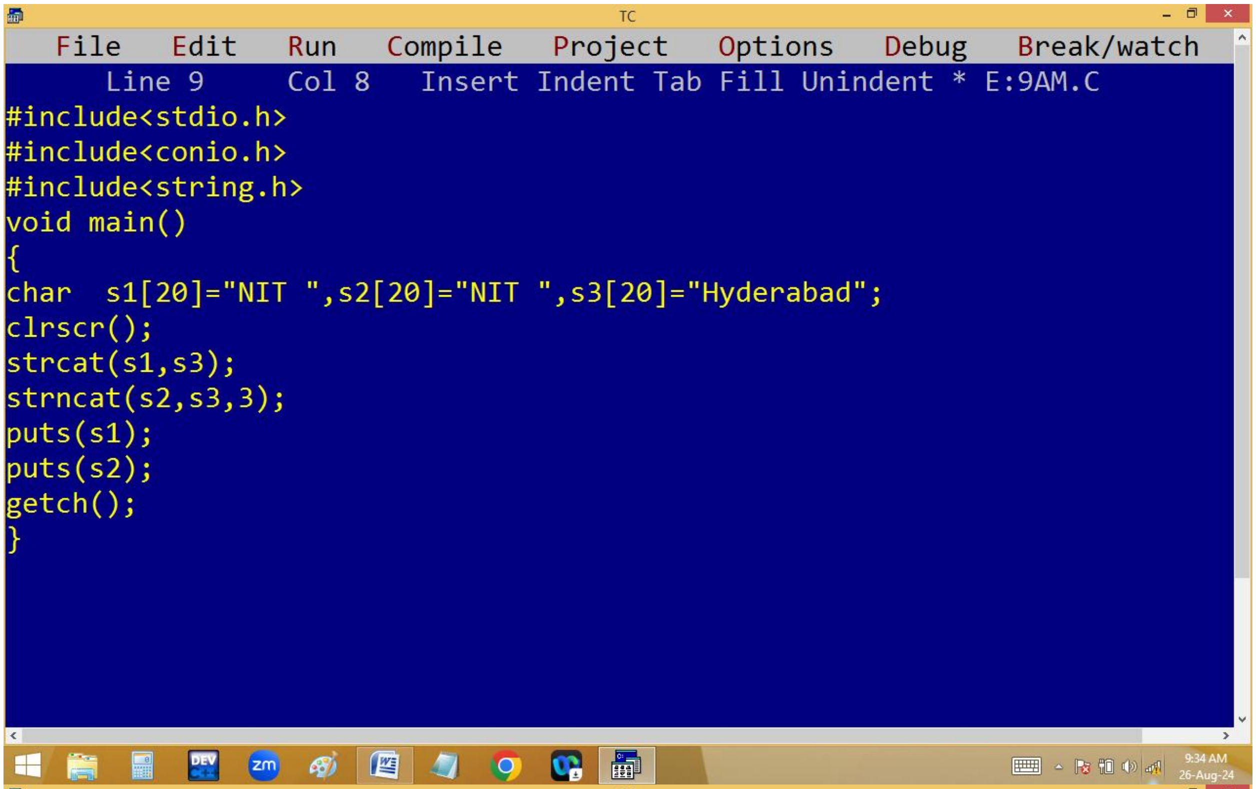
```
Hyderabad
Hyd
```

7. **strcat(): It adds string2 to string1.**

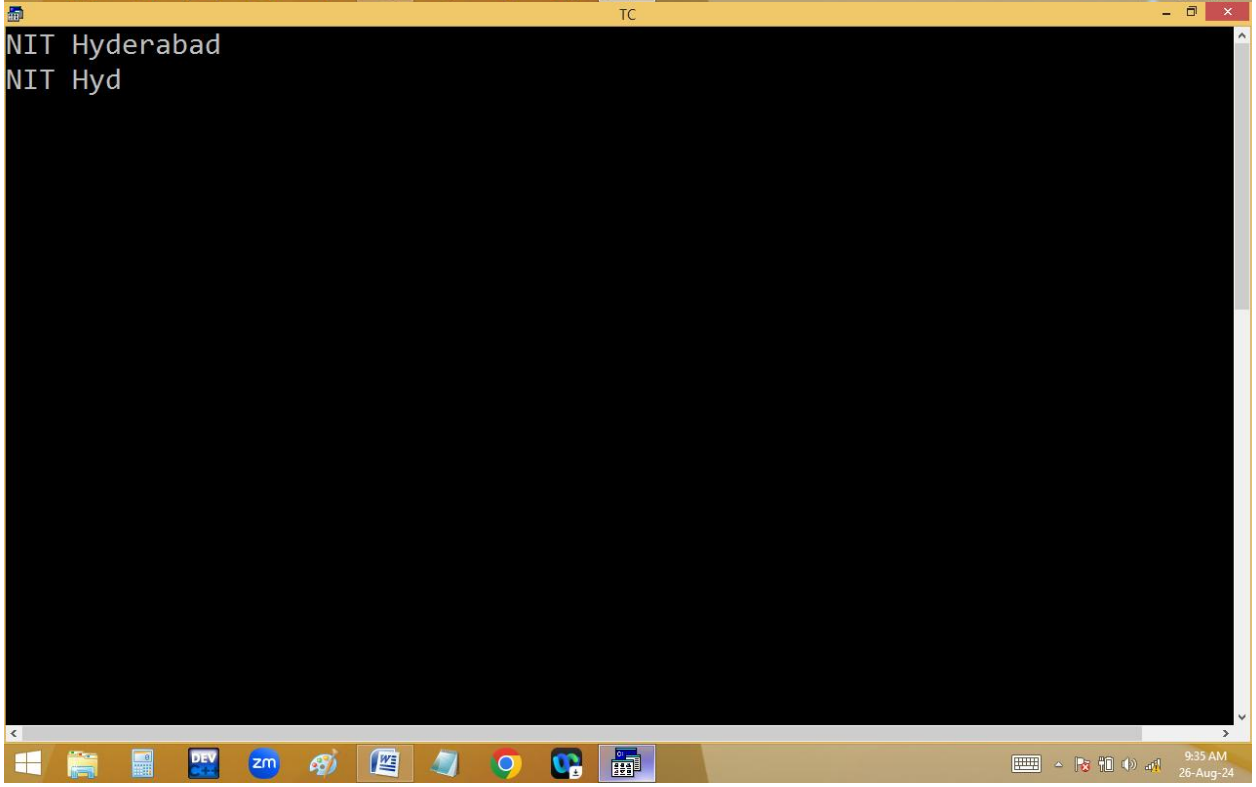
   Syntax: strcat(string1, string2);

8. **strncat(): It adds specified no of char's to string1.**

   Syntax: strncat(string1, string2, no of char);

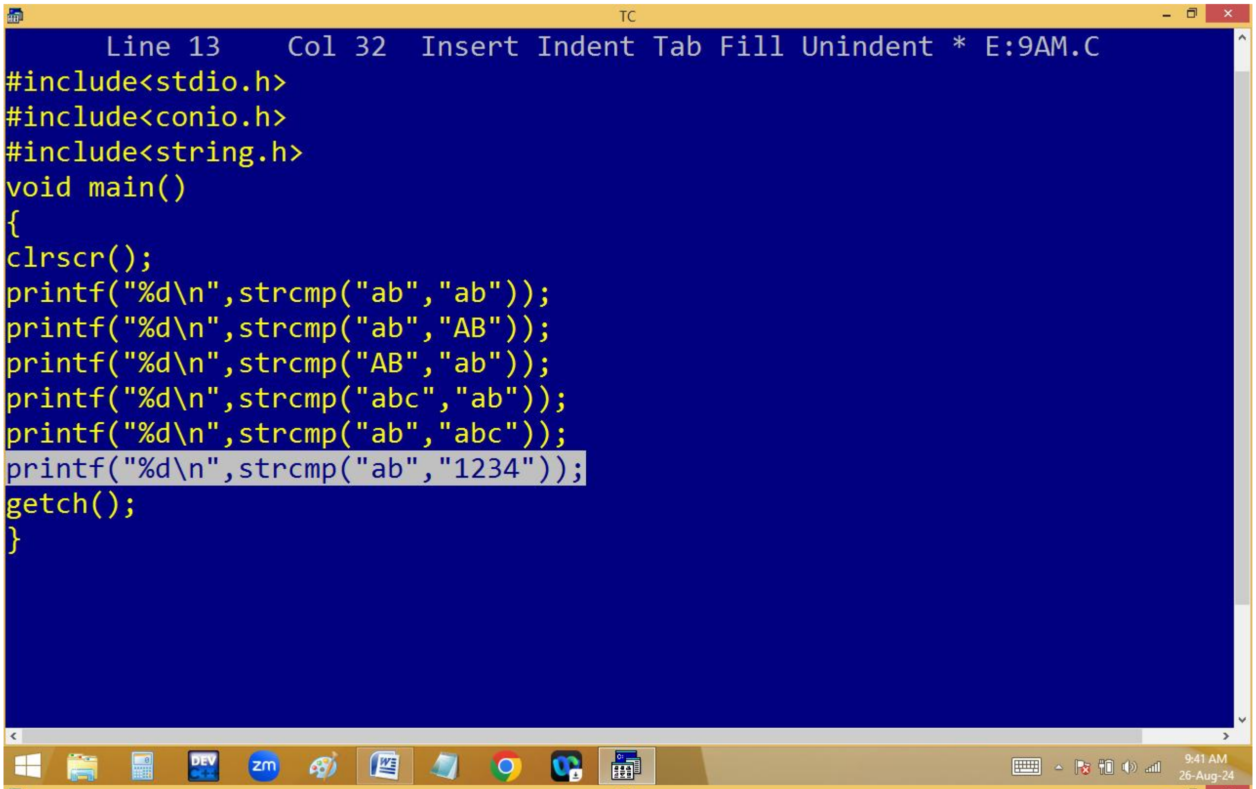

TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 8 Insert Indent Tab Fill Unindent * E:9AM.C
9:34 AM
26-Aug-24

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char  s1[20]="NIT ",s2[20]="NIT ",s3[20]="Hyderabad";
clrscr();
strcat(s1,s3);
strncat(s2,s3,3);
puts(s1);
puts(s2);
getch();
}
```

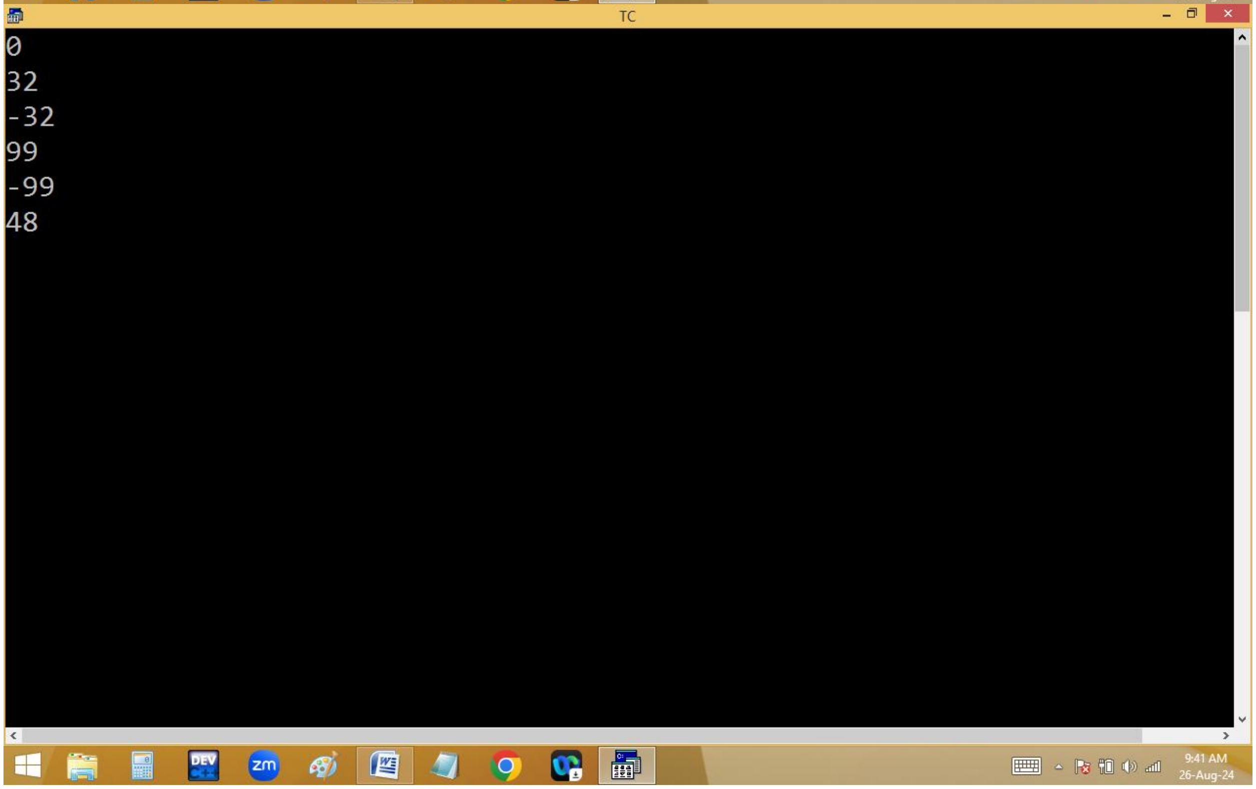


```
NIT Hyderabad
NIT Hyd
```

9. **strcmp(): It compare two strings until the first ascii value difference found.**
   Syntax: strcmp(string1, string2);



```
      Line 13    Col 32  Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
clrscr();
printf("%d\n",strcmp("ab","ab"));
printf("%d\n",strcmp("ab","AB"));
printf("%d\n",strcmp("AB","ab"));
printf("%d\n",strcmp("abc","ab"));
printf("%d\n",strcmp("ab","abc"));
printf("%d\n",strcmp("ab","1234"));
getch();
}
```

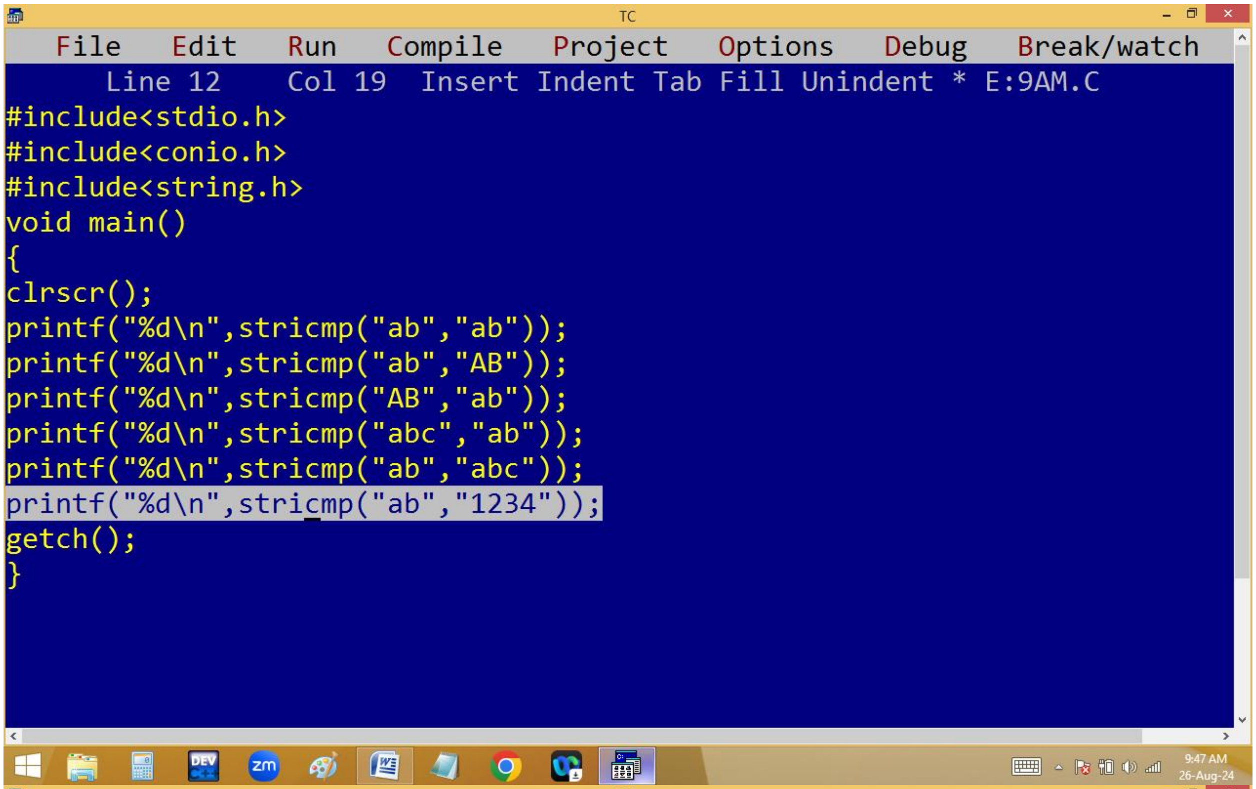


```
0
32
-32
99
-99
48
```

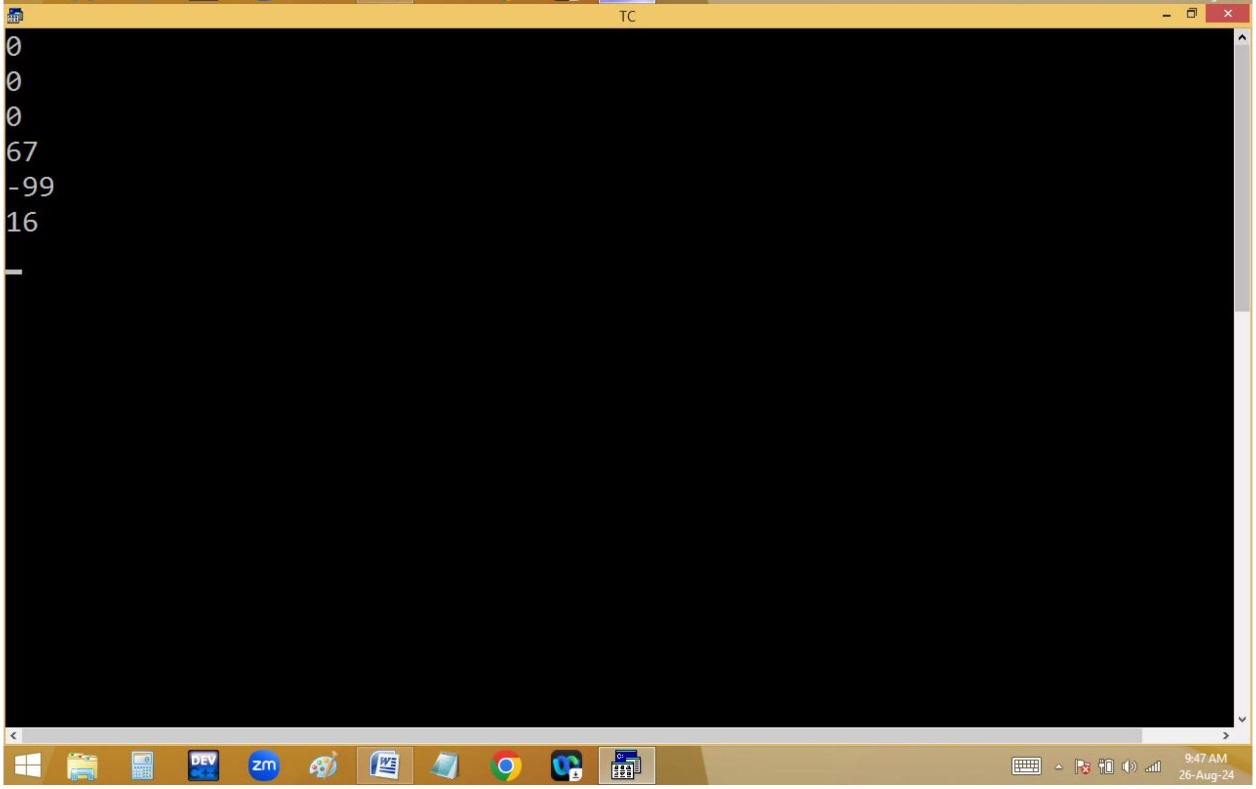
10. **stricmp(): it checks the strings by ignoring case. i.e. in stricmp() lower and upper are same. If matching char not found**

or different data type found in 2nd string, the first string char taken in upper case.

Syntax: stricmp(string1, string2);


TC
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 19 Insert Indent Tab Fill Unindent * E:9AM.C
9:47 AM
26-Aug-24

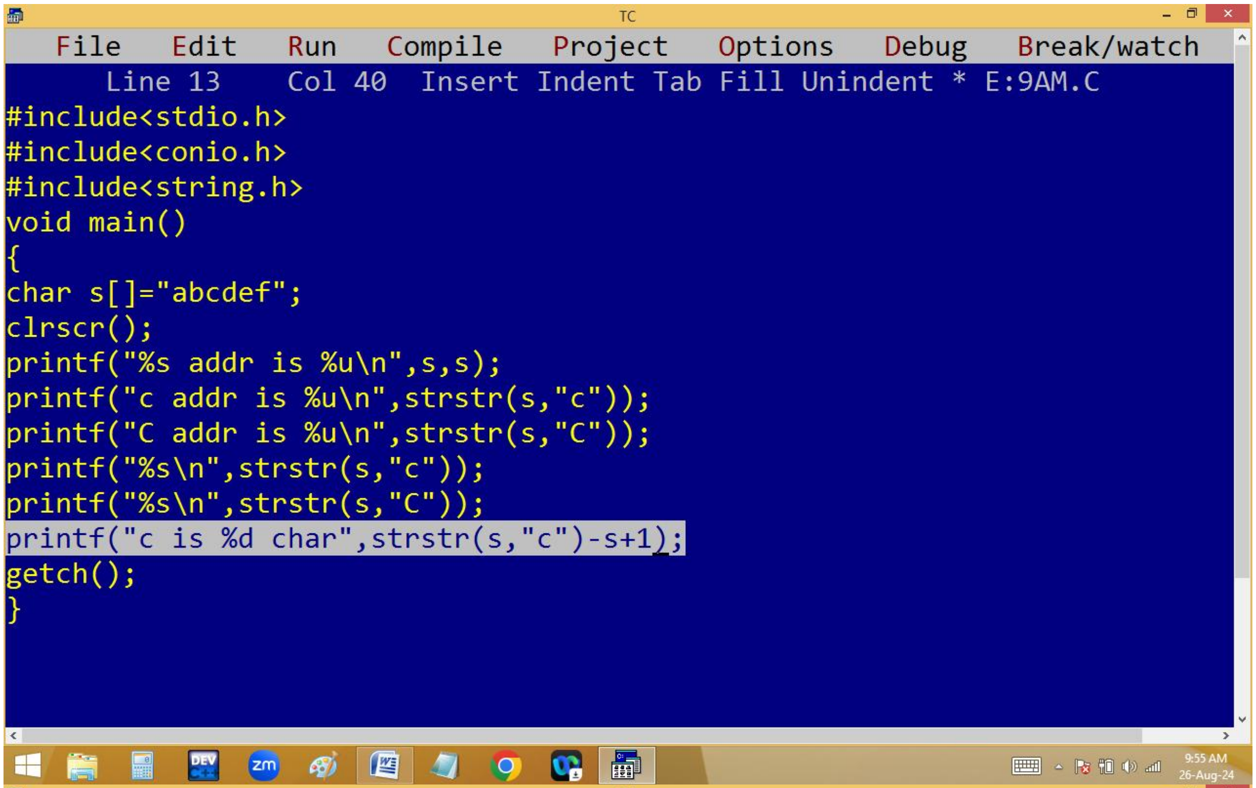
```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
clrscr();
printf("%d\n",stricmp("ab","ab"));
printf("%d\n",stricmp("ab","AB"));
printf("%d\n",stricmp("AB","ab"));
printf("%d\n",stricmp("abc","ab"));
printf("%d\n",stricmp("ab","abc"));
printf("%d\n",stricmp("ab","1234"));
getch();
}
```



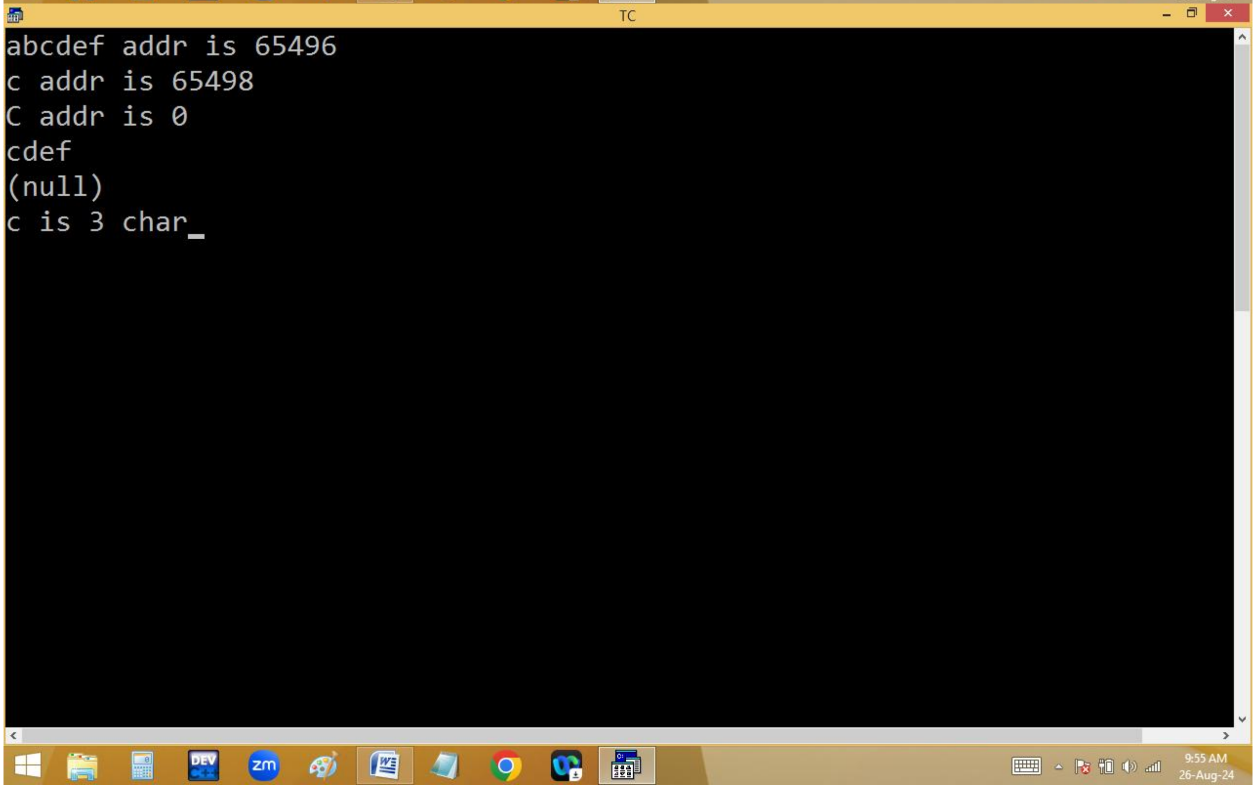
```
0
0
0
67
-99
16
```

11. strstr(): It searches the sub string in main string and if found, it return sub string address. If not found, it return 0.

Syntax: strstr(main string, sub string);

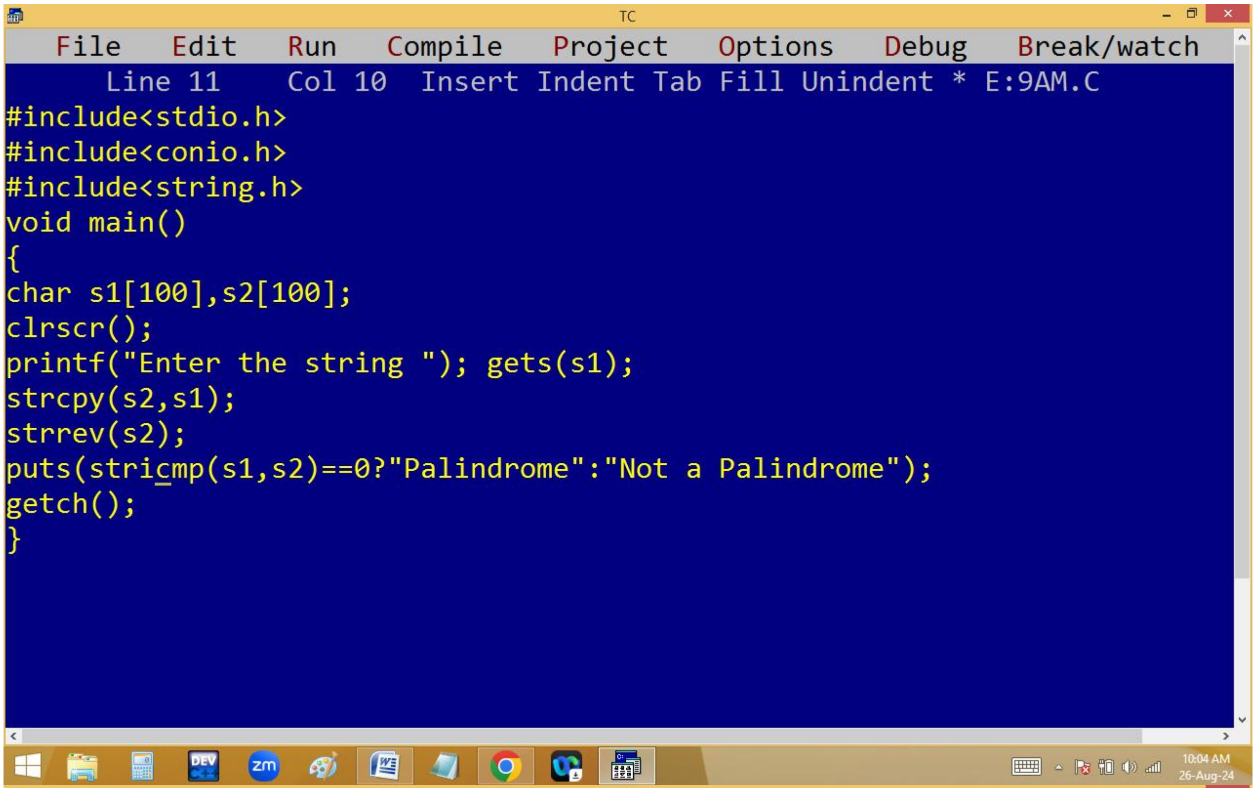


```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[]="abcdef";
clrscr();
printf("%s addr is %u\n",s,s);
printf("c addr is %u\n",strstr(s,"c"));
printf("C addr is %u\n",strstr(s,"C"));
printf("%s\n",strstr(s,"c"));
printf("%s\n",strstr(s,"C"));
printf("c is %d char",strstr(s,"c")-s+1);
getch();
}
```

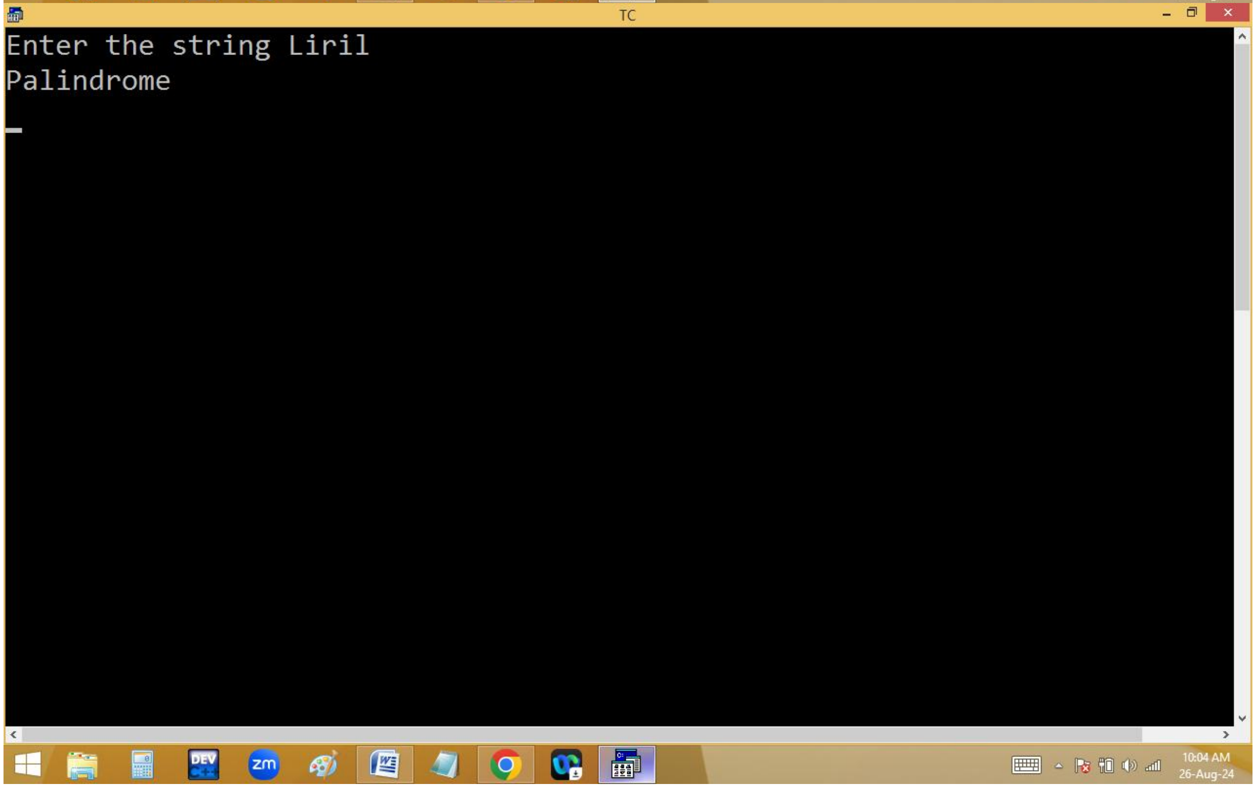


```
abcdef addr is 65496
c addr is 65498
C addr is 0
cdef
(null)
c is 3 char_
```

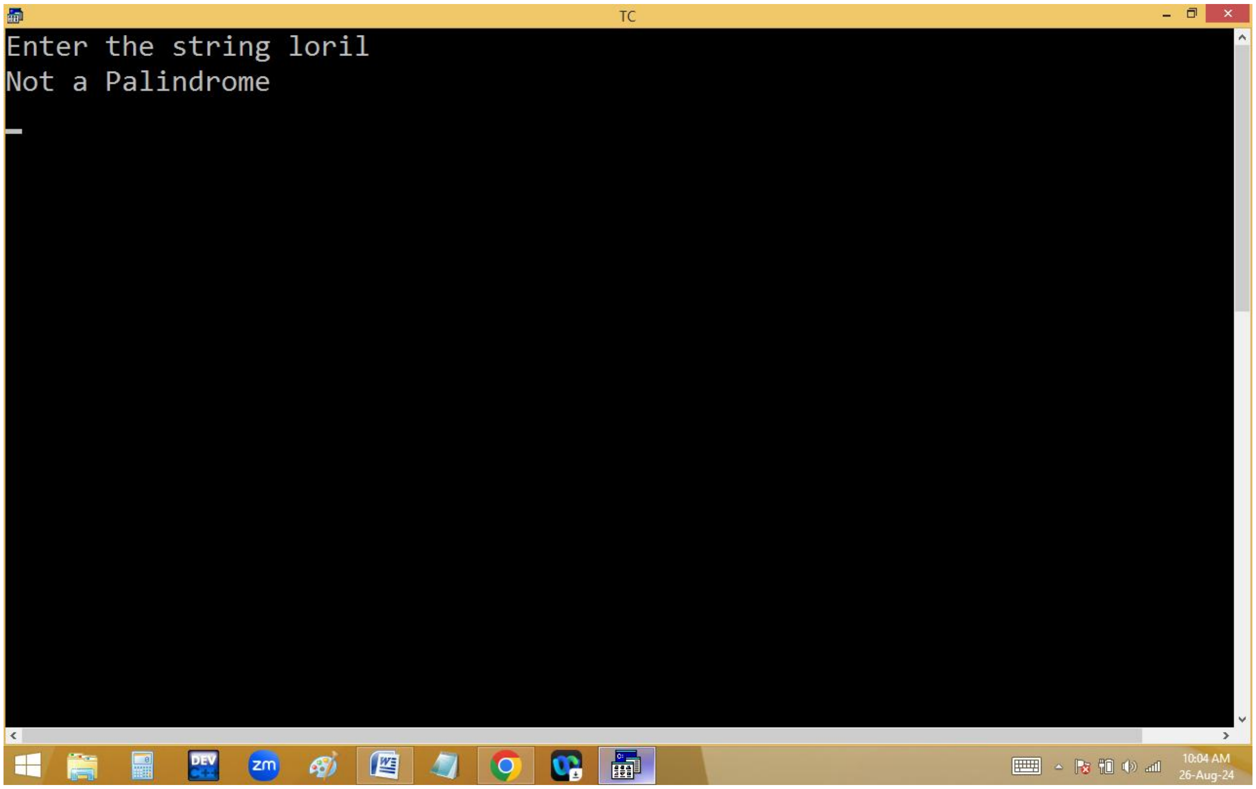
Finding palindrome or not using predefined functions:


TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 10 Insert Indent Tab Fill Unindent * E:9AM.C
10:04 AM
26-Aug-24

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[100],s2[100];
clrscr();
printf("Enter the string "); gets(s1);
strcpy(s2,s1);
strrev(s2);
puts(stricmp(s1,s2)==0?"Palindrome":"Not a Palindrome");
getch();
}
```

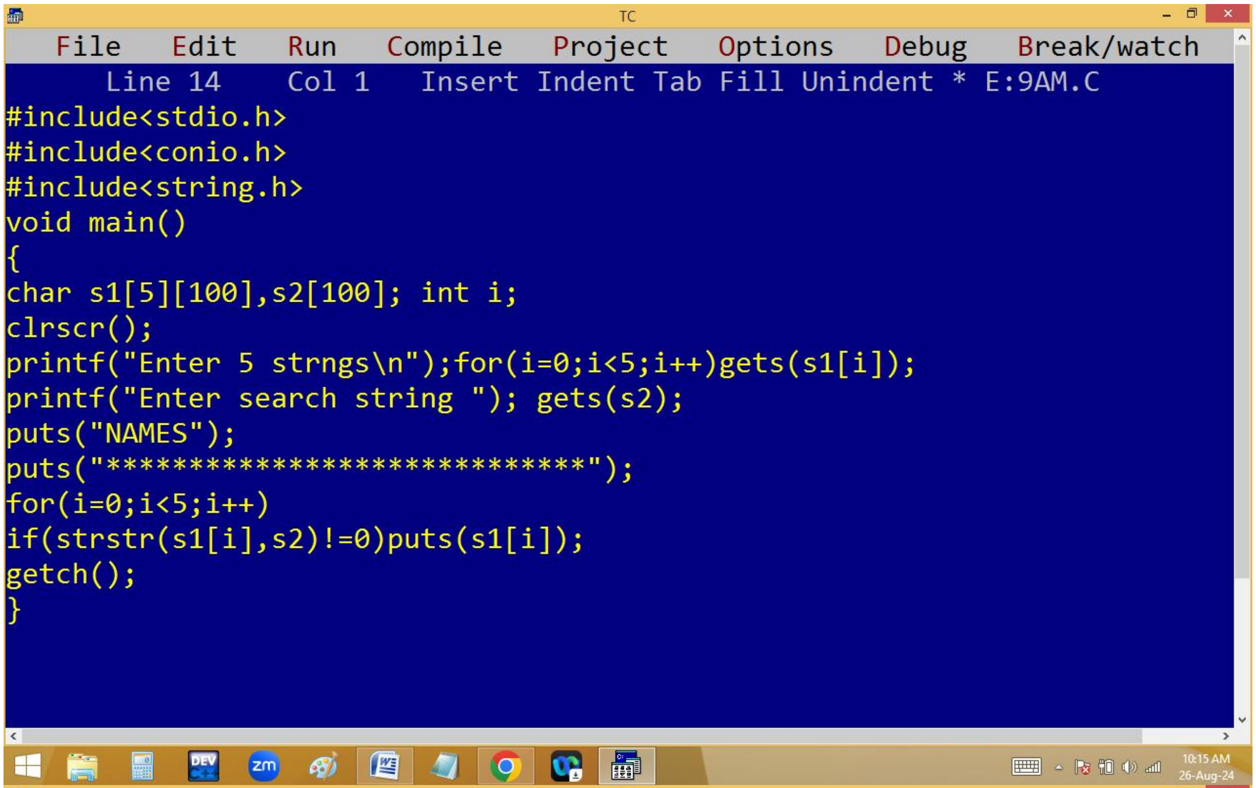


```
Enter the string Liril
Palindrome
```

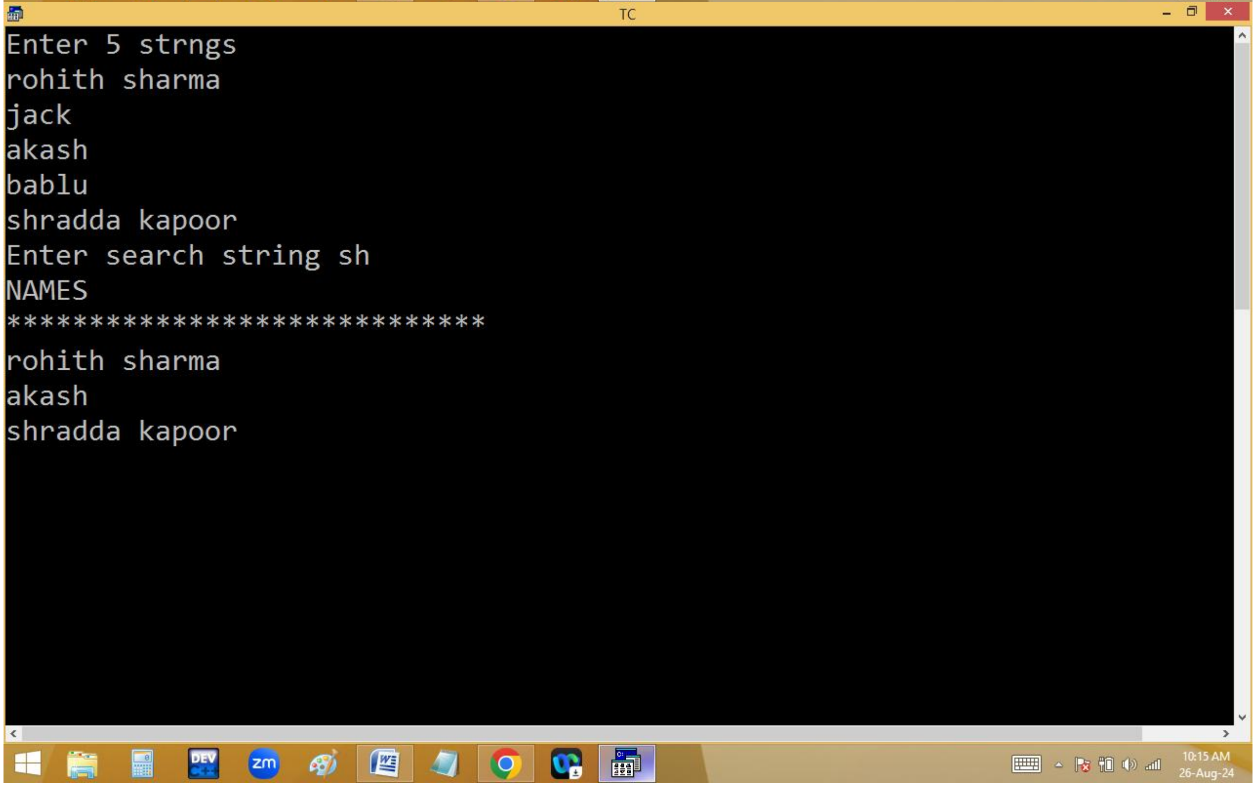


```
Enter the string loril
Not a Palindrome
```

Sorting of strings:

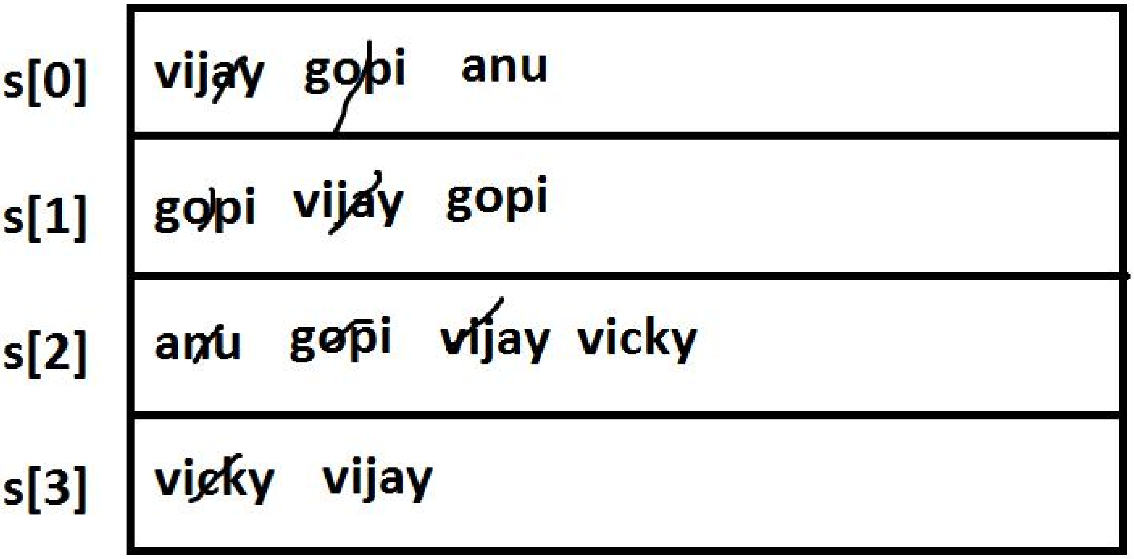


```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[5][100],s2[100]; int i;
clrscr();
printf("Enter 5 strngs\n");for(i=0;i<5;i++)gets(s1[i]);
printf("Enter search string "); gets(s2);
puts("NAMES");
puts("*****************************");
for(i=0;i<5;i++)
if(strstr(s1[i],s2)!=0)puts(s1[i]);
getch();
}
```



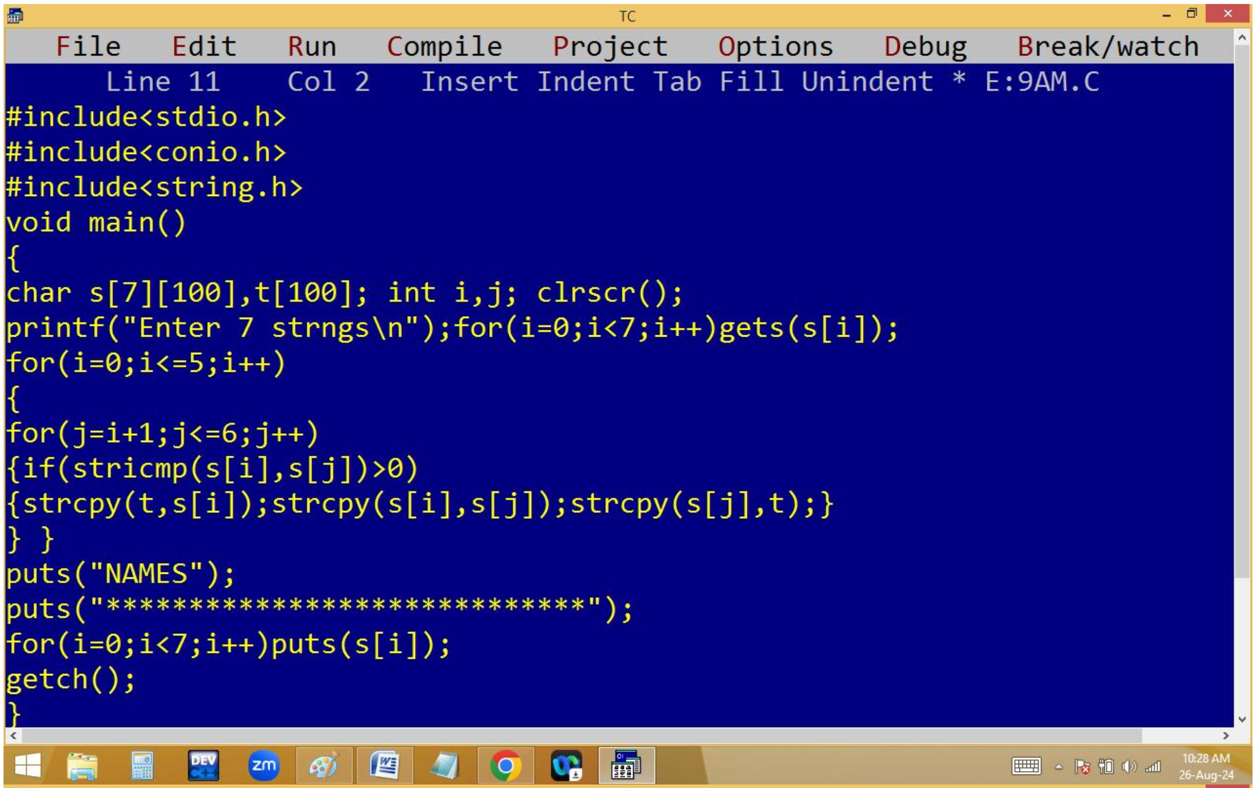
```
Enter 5 strngs
rohith sharma
jack
akash
bablu
shradda kapoor
Enter search string sh
NAMES
*****************************
rohith sharma
akash
shradda kapoor
```

Sorting of strings:

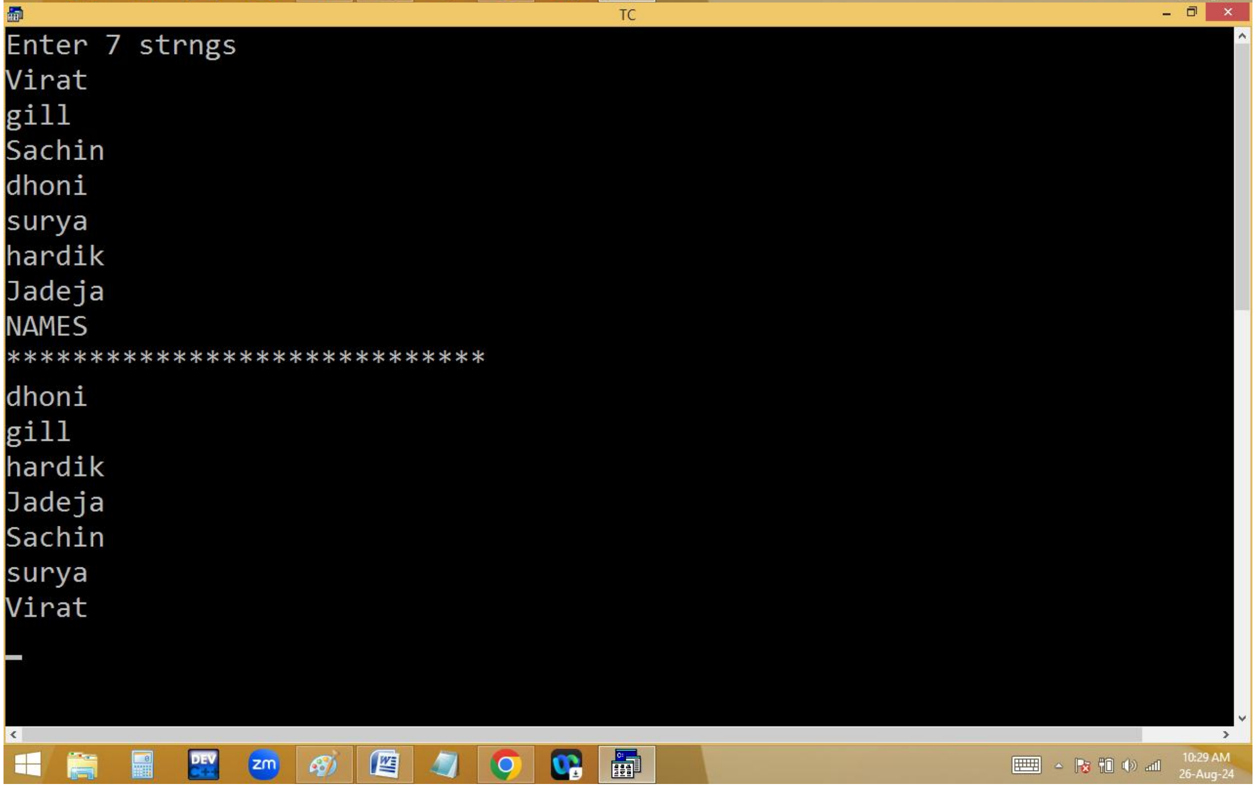


s[0]	~~vijay~~	~~gopi~~	anu	
s[1]	~~gopi~~	~~vijay~~	gopi	
s[2]	~~anu~~	~~gopi~~	~~vijay~~	vicky
s[3]	~~vicky~~	vijay		

selection sort


TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 2 Insert Indent Tab Fill Unindent * E:9AM.C
10:28 AM
26-Aug-24

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[7][100],t[100]; int i,j; clrscr();
printf("Enter 7 strngs\n");for(i=0;i<7;i++)gets(s[i]);
for(i=0;i<=5;i++)
{
for(j=i+1;j<=6;j++)
{if(stricmp(s[i],s[j])>0)
{strcpy(t,s[i]);strcpy(s[i],s[j]);strcpy(s[j],t);}
} }
puts("NAMES");
puts("*****************************");
for(i=0;i<7;i++)puts(s[i]);
getch();
}
```



```
Enter 7 strngs
Virat
gill
Sachin
dhoni
surya
hardik
Jadeja
NAMES
*****************************
dhoni
gill
hardik
Jadeja
Sachin
surya
Virat
_
```

POINTERS:

Pointer is a variable which stores the address of another variable of same type.

Pointer is a variable which stores the address of memory [bytes] at runtime [dynamic].

```
char  name[20];
```