

Captcha code generation:

```
#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void main()

{

char s[10],sp[]="@#$%&*",ch; int n,i;

clrscr();

while(1)

{randomize();

for(i=0;i<5;i++)

{

n=random(4);

if(n==0)s[i]=random(26)+97;

else if(n==1)s[i]=random(26)+65;

else if(n==2)s[i]=random(10)+48;

else s[i]=sp[random(6)];
```

```
}s[i]='\0';  
printf("Captcha = %s\n",s);  
flushall(); printf("Refresh [y/n] ");scanf("%c",&ch);  
if(ch=='n' || ch=='N')break;  
  
}  
  
}
```

```
TC
Captcha = teJ*w
Refresh [y/n] y
Captcha = f%265
Refresh [y/n] y
Captcha = 91lk*
Refresh [y/n] n
```

```
while(1)
{
for(i=0;i<5;i++)
{
n=random(4);
if(n==0)s[i]=random(26)+97;
else if(n==1)s[i]=random(26)+65;
else if(n==2)s[i]=random(10)+48;
else s[i]=sp[random(6)];
} s[i]='\0';
p(s);
}
```

```
p("Refresh [y/n]");
s("%c",&ch);
if(ch=='n' || ch=='N')
break;
}
```

d	Z	5	\$	A	\0	
0	1	2	3	4	5	6

- 0-lower
- 1-upper
- 2-digits
- 3-special - @#\$%*&

O T P Generation:

```
TC
Line 18 Col 12 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
char s[10],ch; int i;
clrscr();
while(1)
{randomize();
for(i=0;i<8;i++)
{
s[i]=random(10)+48;
}s[i]='\0';
printf("O T P = %s\n",s);
flushall(); printf("Resend O T P [y/n] ");scanf("%c",&ch);
if(ch=='n' || ch=='N')break;
}
}
```

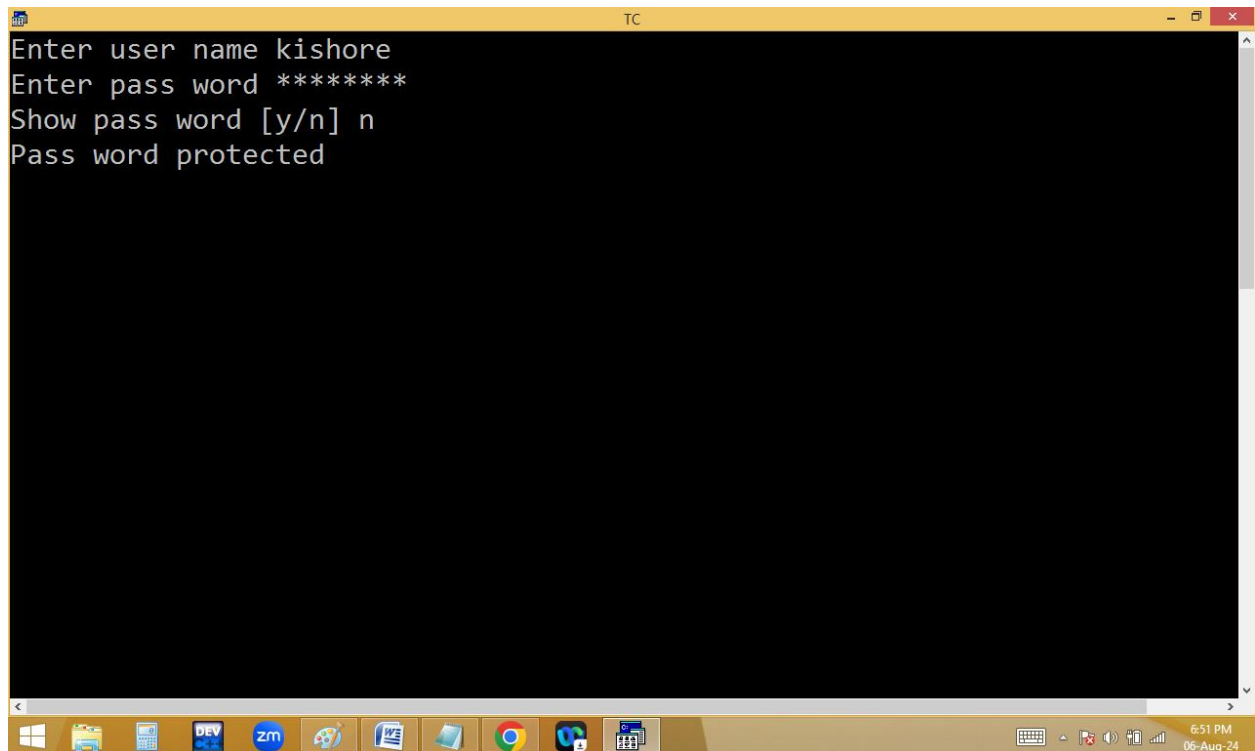
```
O T P = 23483062
Resend O T P [y/n] y
O T P = 80498207
Resend O T P [y/n] y
O T P = 09052485
Resend O T P [y/n] n
```

Password Generation:

The image shows a screenshot of a Turbo C++ (TC) IDE. The top window displays the source code for a program that prompts the user for a username and password, then asks if they want to show the password. The bottom window shows the program's output, where the user has entered 'kishore' as the username, '*****' as the password, and 'y' to show the password, resulting in the output 'Ur pass word is micromax'.

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 26 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char user[20],pass[20],ch; int i=0; clrscr();
printf("Enter user name "); scanf("%s",user);
printf("Enter pass word ");
while((ch=getch())!=13)
{
printf("*"); pass[i]=ch; i++;
}
pass[i]='\0';
printf("\nShow pass word [y/n] "); fflush(); scanf("%c",&ch);
if(ch=='y' || ch=='Y') printf("Ur pass word is %s",pass);
else puts("Pass word protected");
getch();
}
```

Enter user name kishore
Enter pass word *****
Show pass word [y/n] y
Ur pass word is micromax



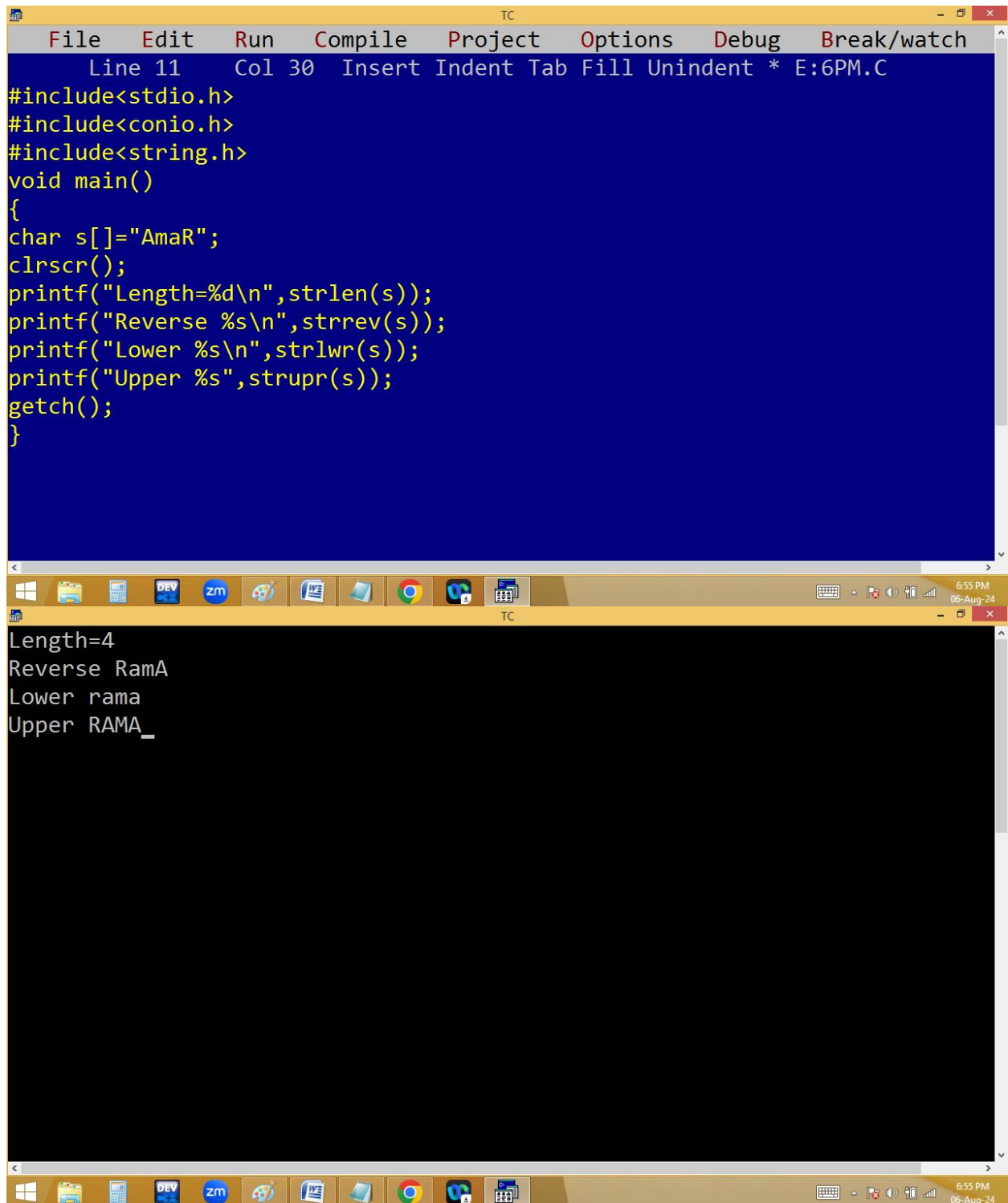
```
TC
Enter user name kishore
Enter pass word *****
Show pass word [y/n] n
Pass word protected
```

STRING LIBRARY FUNCTIONS

To manage string operations c provides several predefined functions available in string.h

They are

1. strlen(): Return the no of char's in given string.
2. strrev(): return reverse string
3. strlwr(): converts into lower case.
- 4.strupr(): converts into upper case.



The image shows a screenshot of the Turbo C++ (TC) IDE. The top window displays a C program with the following code:

```
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 30 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[]="AmaR";
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();
}
```

The bottom window shows the output of the program:

```
Length=4
Reverse RamA
Lower rama
Upper RAMA_
```

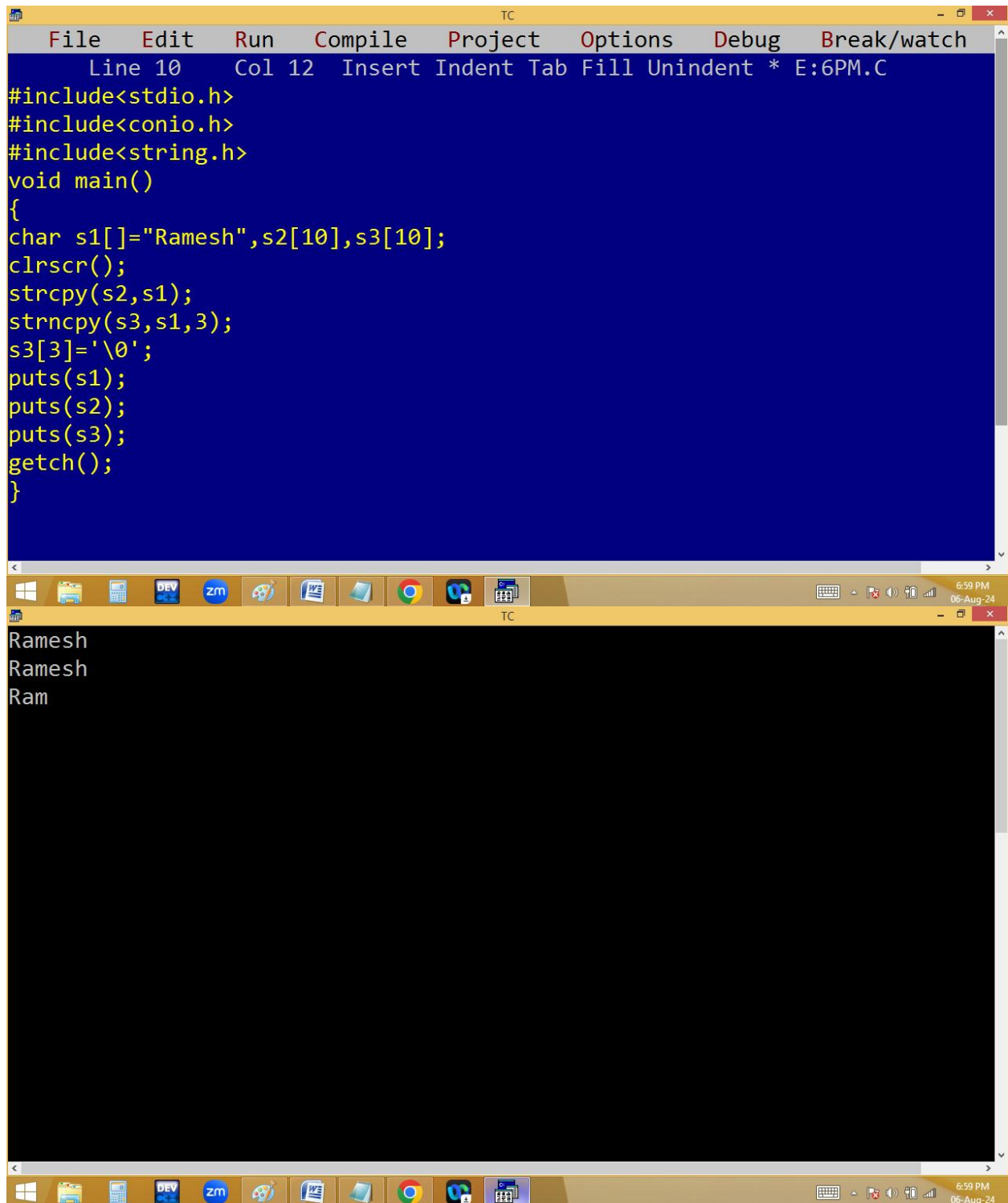
The IDE interface includes a menu bar at the top with options: File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the bottom of the output window shows the time as 6:55 PM on 06-Aug-24.

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

Strcpy(destination string, source string);

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

Strncpy(deststring, sourcestring, no of char's);



The screenshot shows the Turbo C++ (TC) IDE. The top window displays the source code for a C program. The code includes headers for `stdio.h`, `conio.h`, and `string.h`. The `main` function initializes three character arrays: `s1` with "Ramesh", `s2` with 10 zeros, and `s3` with 10 zeros. It then prints `s1`, `s2`, and `s3` using `puts`, and waits for a key press with `getch`. The bottom window shows the output of the program, which displays "Ramesh", "Ramesh", and "Ram" (the first three characters of `s3`) on separate lines. The Windows taskbar at the bottom shows the time as 6:59 PM on 06-Aug-24.

```
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 12 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[]="Ramesh",s2[10],s3[10];
clrscr();
strcpy(s2,s1);
strncpy(s3,s1,3);
s3[3]='\0';
puts(s1);
puts(s2);
puts(s3);
getch();
}
```

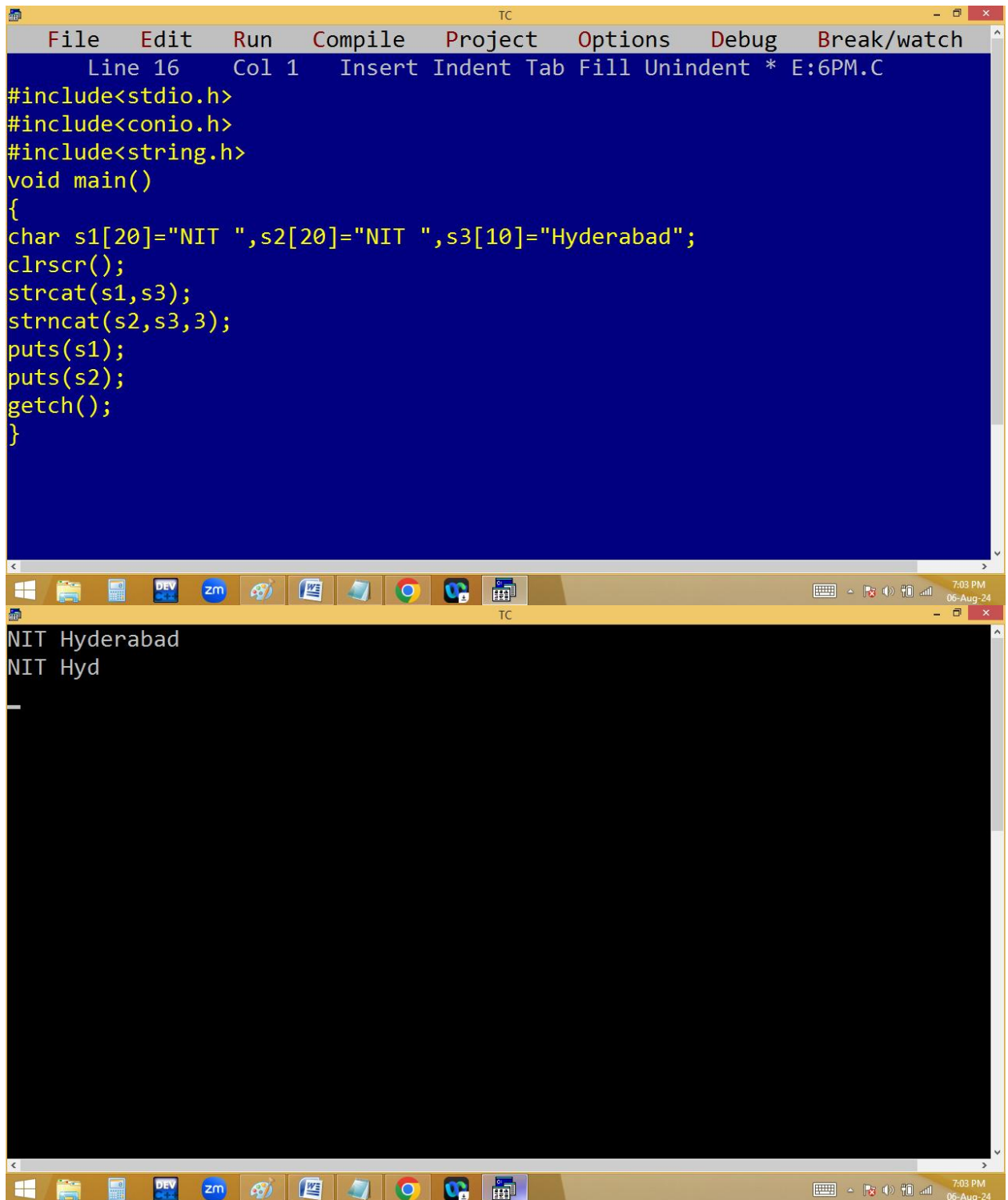
Ramesh
Ramesh
Ram

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

Strcat(string1, string2);

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

Strncat(string1, string2, no of char's);



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 16 Col 1 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[20]="NIT ",s2[20]="NIT ",s3[10]="Hyderabad";
clrscr();
strcat(s1,s3);
strncat(s2,s3,3);
puts(s1);
puts(s2);
getch();
}
```

NIT Hyderabad
NIT Hyd

9. **strcmp()**: It compare two string using ascii values and return the first ascii difference.

```
Strcmp(string1, string2);
```

The screenshot displays the Turbo C++ (TC) IDE. The top window shows a C program with the following code:

```
File Edit Run Compile Project Options Debug Break/watch
Line 13 Col 32 Insert Indent Tab Fill Unindent * E:6PM.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();
}
```

The bottom window shows the output of the program:

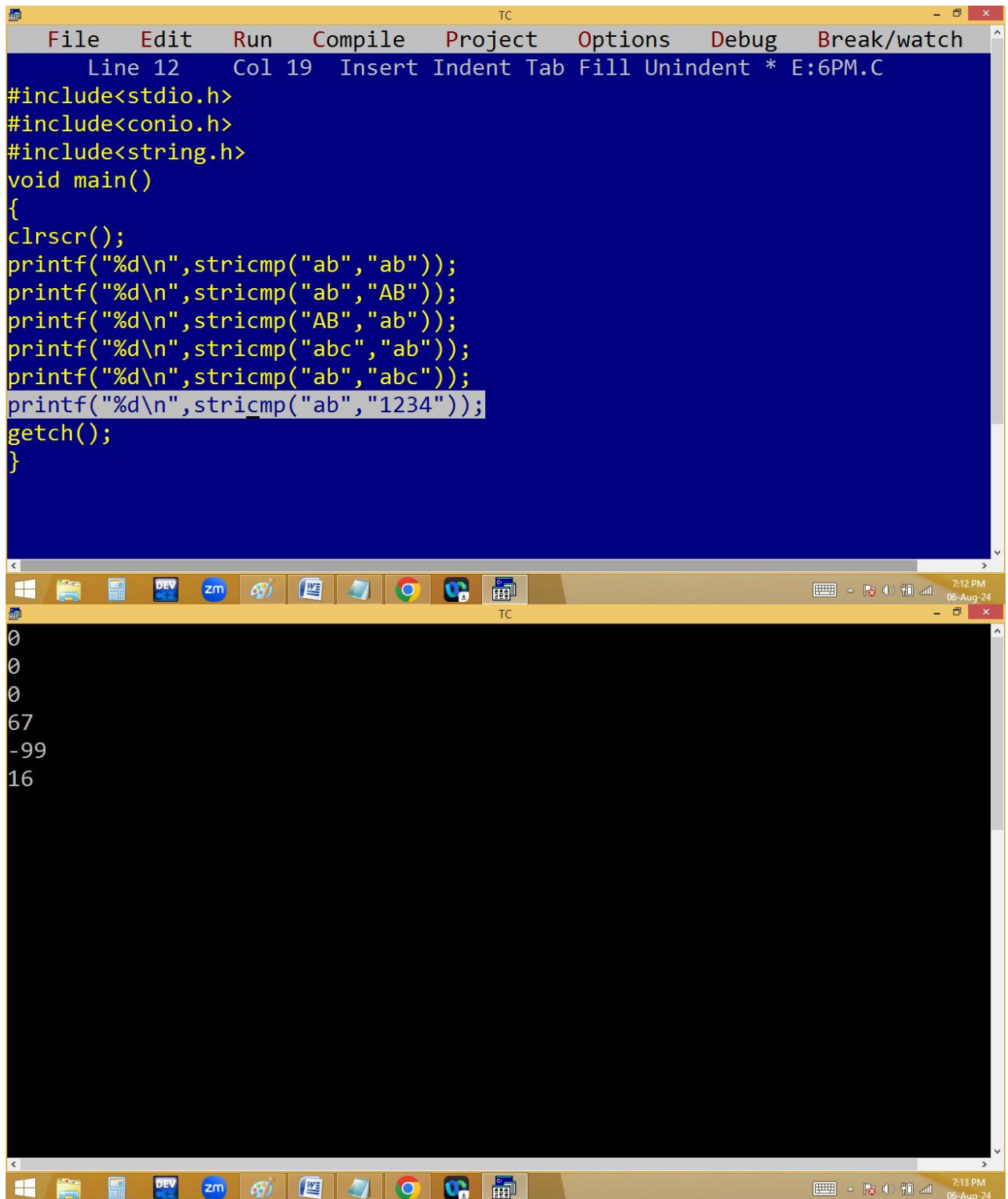
```
0
32
-32
99
-99
48
```

The output values correspond to the strcmp() results for the following comparisons: "ab" vs "ab" (0), "ab" vs "AB" (32), "AB" vs "ab" (-32), "abc" vs "ab" (99), "ab" vs "abc" (-99), and "ab" vs "1234" (48).

10. **stricmp()**: it compare two strings by ignoring the case. i.e. in stricmp() lower and upper are same.

When matching char not found or different data type found in 2nd string, the 1st string char taken in upper case.

Stricmp(string1, string2);



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 19 Insert Indent Tab Fill Unindent * E:6PM.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
0
0
67
-99
16

11. **strstr()**: it return the sub string address in main string. If sub string is not found, it return 0.
Strstr(main string, sub string);

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

abcdef address is 65496
c address is 65498
cdef
C address is 0
(null)
c is 3 char in abcdef

TC

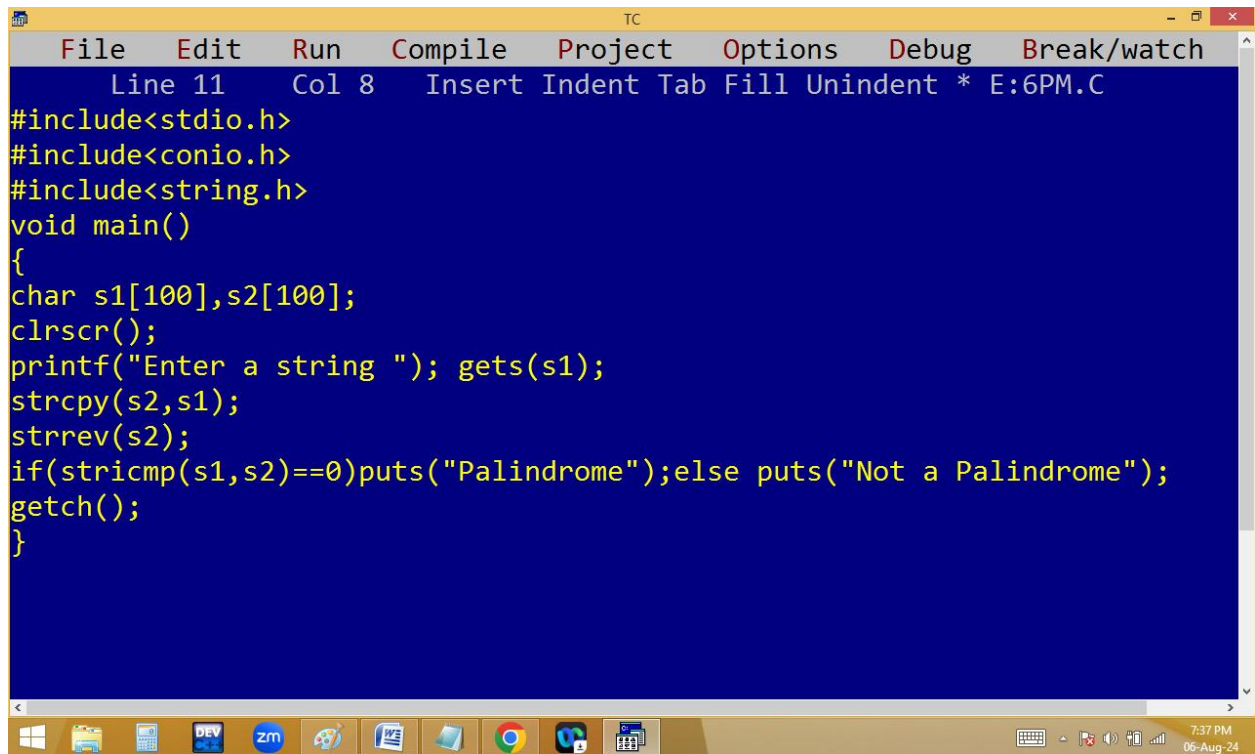
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 3 Col 1 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[5][20],sub[2]; int i;
clrscr();
puts("Enter 5 names "); for(i=0;i<5;i++)gets(s[i]);
printf("Enter the sub string to search "); gets(sub);
puts("NAMES");
puts("-----");
for(i=0;i<5;i++)if(strstr(s[i],sub)!=0)puts(s[i]);
getch();
}

Enter 5 names
himanshu
bablu
shashank
akshay
prem
Enter the sub string to search sh
NAMES
-----
himanshu
shashank
akshay

```


Finding palindrome using string library

functions:



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 8 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[100],s2[100];
clrscr();
printf("Enter a string "); gets(s1);
strcpy(s2,s1);
strrev(s2);
if(strcmp(s1,s2)==0)puts("Palindrome");else puts("Not a Palindrome");
getch();
}
```

The screenshot shows a Turbo C++ IDE window titled 'TC'. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 11 Col 8 Insert Indent Tab Fill Unindent * E:6PM.C'. The main editing area has a blue background and contains the following C code: `#include<stdio.h>`, `#include<conio.h>`, `#include<string.h>`, `void main()`, `{`, `char s1[100],s2[100];`, `clrscr();`, `printf("Enter a string "); gets(s1);`, `strcpy(s2,s1);`, `strrev(s2);`, `if(strcmp(s1,s2)==0)puts("Palindrome");else puts("Not a Palindrome");`, `getch();`, and `}`. The Windows taskbar at the bottom shows various application icons and the system clock indicating 7:37 PM on 06-Aug-24.

```
TC
Enter a string Akka
Palindrome
```

```
TC
Enter a string anil
Not a Palindrome
```

Sorting of strings:

s[0]	p rem g opi bablu
s[1]	g opi p rem g opi gita
s[2]	b ablu g opi p rem gopi
s[3]	g ita gopi prem

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<string.h>
```

```
void main()
```

```
{
```

```
char s[7][100],t[100]; int i,j;
```

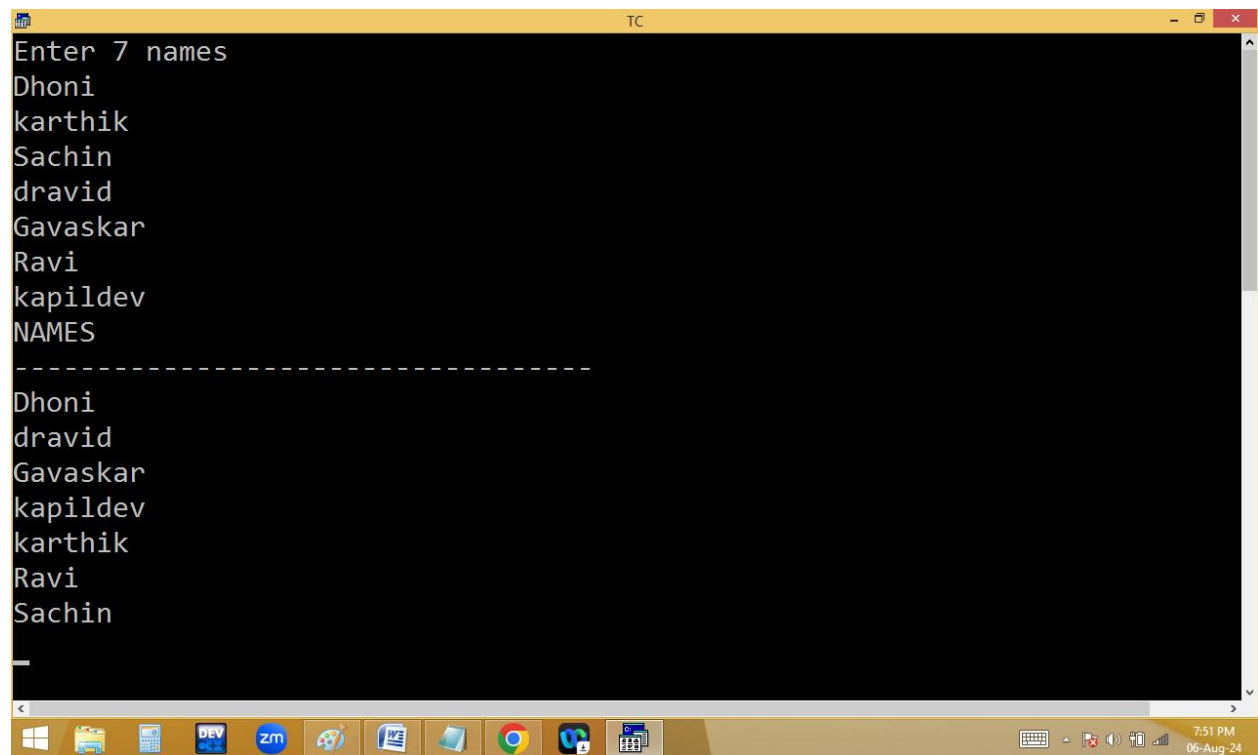
```
clrscr();
```

```
printf("Enter 7 names\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();  
}
```

```
TC
Enter 7 names
Dhoni
karthik
Sachin
dravid
Gavaskar
Ravi
kapildev
NAMES
-----
Dhoni
dravid
Gavaskar
kapildev
karthik
Ravi
Sachin
_
```



The image shows a Windows 10 desktop environment. A terminal window titled 'TC' is open, displaying a program that prompts the user to 'Enter 7 names'. The user has entered the following names: Dhoni, karthik, Sachin, dravid, Gavaskar, Ravi, and kapildev. The program then prints the names in reverse order, separated by a dashed line. The taskbar at the bottom shows various application icons, including Windows Explorer, DEV, zm, and Google Chrome. The system clock in the bottom right corner indicates the time is 7:51 PM on 06-Aug-24.