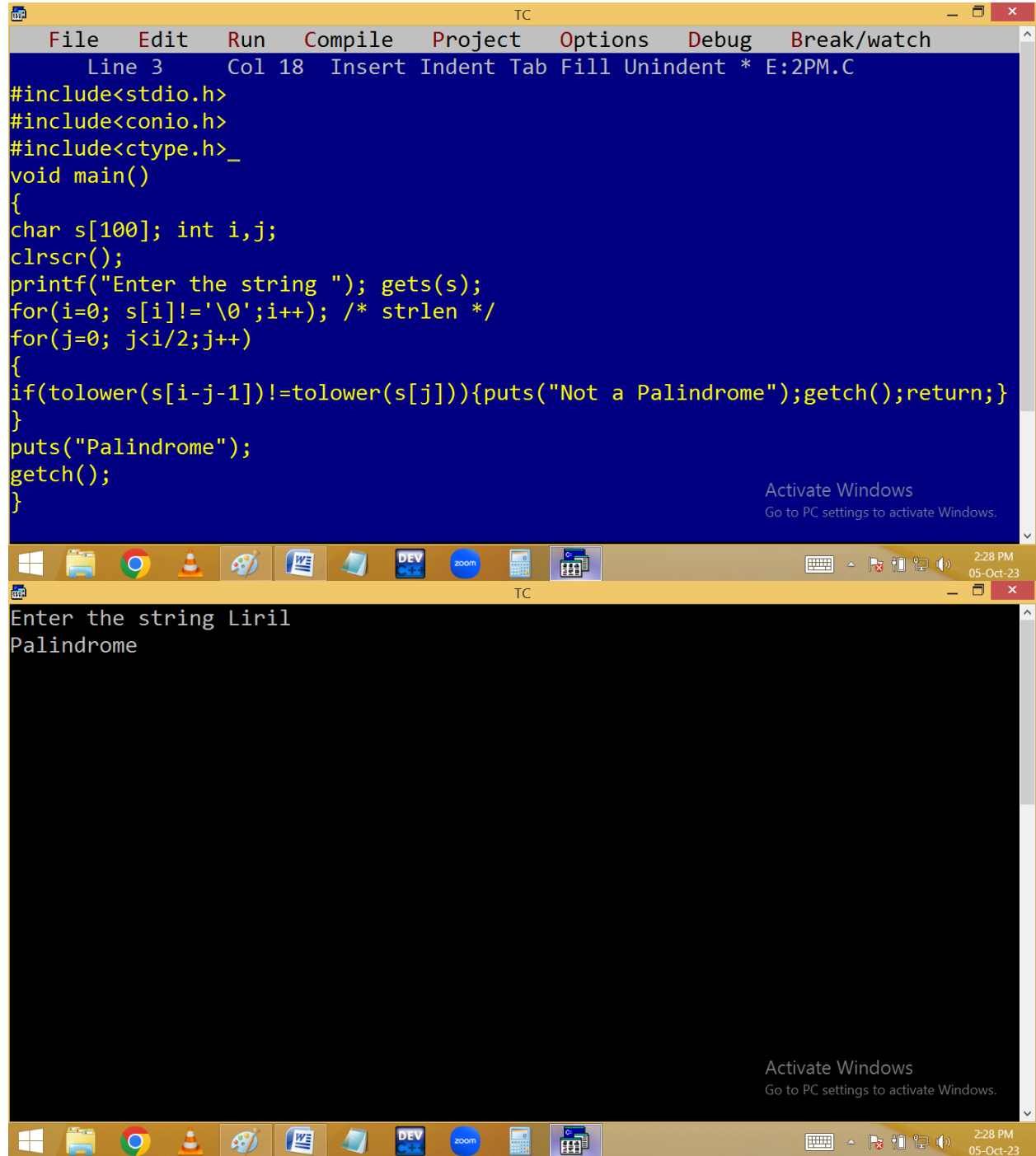


Finding palindrome or not using single string:



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 3 Col 18 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<ctype.h>
void main()
{
char s[100]; int i,j;
clrscr();
printf("Enter the string "); gets(s);
for(i=0; s[i]!='\0';i++); /* strlen */
for(j=0; j<i/2;j++)
{
if(tolower(s[i-j-1])!=tolower(s[j])){puts("Not a Palindrome");getch();return;}
}
puts("Palindrome");
getch();
}
```

Activate Windows
Go to PC settings to activate Windows.

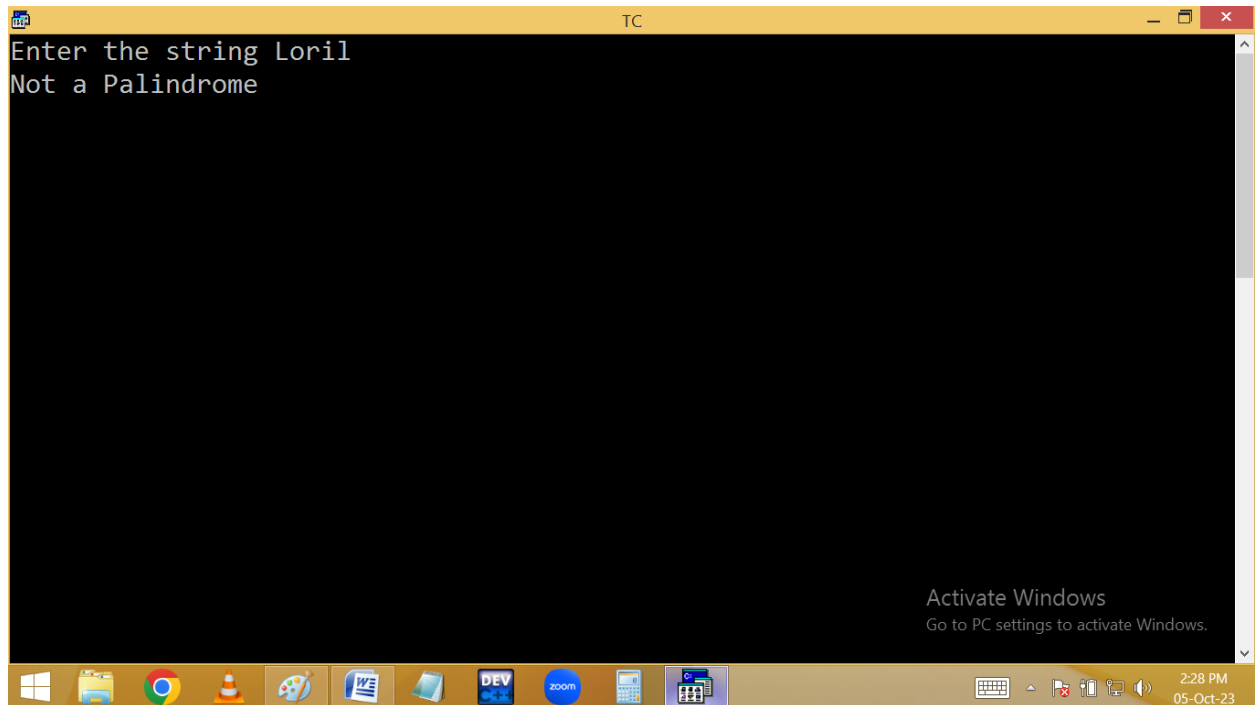
2:28 PM
05-Oct-23

TC

Enter the string Liril
Palindrome

Activate Windows
Go to PC settings to activate Windows.

2:28 PM
05-Oct-23



```
for( i=0; s[i]!='\0';i++) ; /* strlen */
```

```
for( j=0; j<i/2; j++)
```

```
{
```

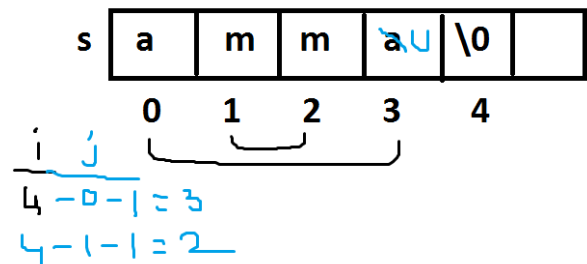
```
if(s[j] != s[i-j-1] )
```

```
{ p("Not a palindrome");
```

```
    getch(); return; }
```

```
} p("Palindrome");
```

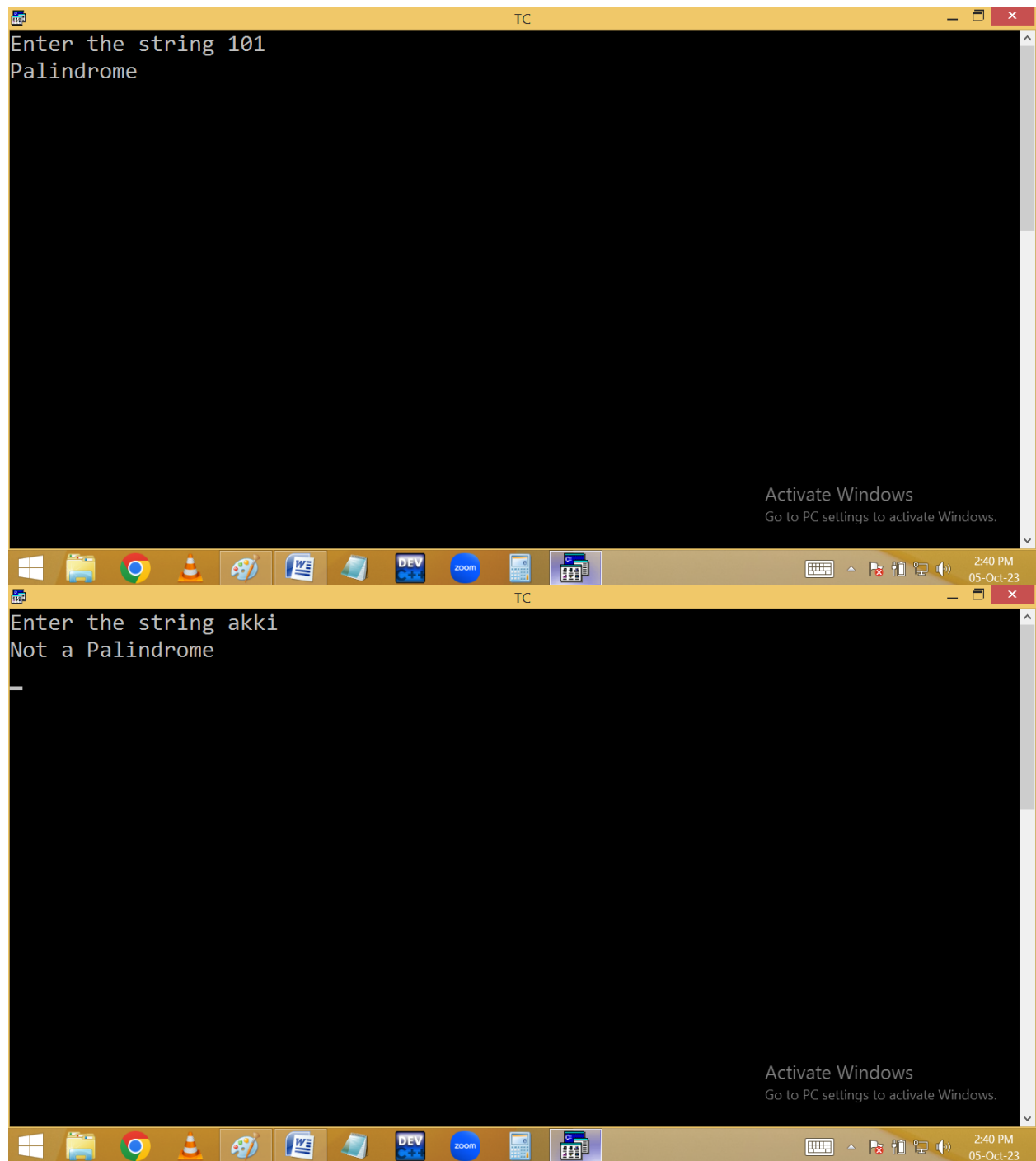
j
0
1



The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code for a C program that checks if a string is a palindrome. The code includes headers for `stdio.h`, `conio.h`, and `ctype.h`. It defines a `main` function that reads a string `s1` and copies it in reverse order to `s2`. It then compares the two strings character by character (case-insensitive) to determine if it's a palindrome. The bottom screenshot shows the program's execution. It prompts the user to "Enter the string" and the input "malayalam" is provided. The program outputs "Palindrome".

```
File Edit Run Compile Project Options Debug Break/watch
Line 13 Col 32 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<ctype.h>
void main()
{
char s1[100],s2[100]; int i,j;
clrscr();
printf("Enter the string "); gets(s1);
for(i=0; s1[i]!='\0';i++); /* strlen */
for(s2[i--]='\0', j=0; i>=0;j++,i--)s2[j]=s1[i]; /* rev copy */
for(i=0;s1[i]!='\0';i++)
{
if(tolower(s1[i])!=tolower(s2[i])){puts("Not a Palindrome");getch();return;}
}
puts("Palindrome");
getch();
}
```

Enter the string malayalam
Palindrome



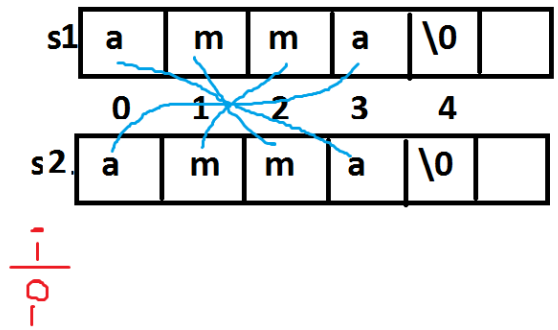
```

for( i=0; s[i]!='\0';i++) ; /* strlen */
s2[i--]='\0';
for( j=0; j>=0; j++,i--)
s2[j]=s1[i]; /* rev copy */
for(i=0; s1[i]!='\0';i++)
{
if(s1[i]!=s2[i]) /* strcmp */
{p("Not");getch();return;}
} p(palindrome);

```

Handwritten annotations for the code above:

- Under `j=0`: $\frac{i}{4} \rightarrow \frac{j}{3}$
- Under `j++`: $2 \rightarrow 1$
- Under `i--`: $1 \rightarrow 2$
- Under `s1[i]`: $0 \rightarrow 3$
- Under `s2[i]`: $-1 \rightarrow 4$



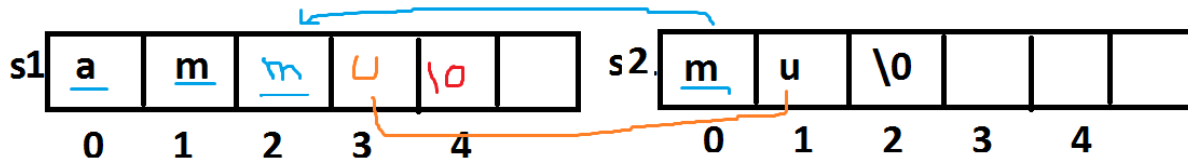
String concatenation [Adding of two strings]:

The image shows two windows of the Turbo C++ (TC) IDE. The top window displays the source code for a C program named E:2PM.C. The code includes `<stdio.h>` and `<conio.h>`, and defines a `main` function. Inside `main`, it declares two character arrays `s1` and `s2` of size 100, and two integer variables `i` and `j`. It calls `clrscr()` to clear the screen. It then prompts the user to enter a first name and a last name using `gets()`. A loop concatenates the last name into the first name array `s1` character by character. Finally, it prints the concatenated string and waits for a key press with `getch()`.

```
File Edit Run Compile Project Options Debug Break/watch
Line 3 Col 1 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s1[100],s2[100]; int i,j;
clrscr();
printf("Enter first name "); gets(s1);
printf("Enter last name "); gets(s2);
for(i=0; s1[i]!='\0';i++); /* strlen */
for( j=0; s2[j]!='\0';j++,i++)s1[i]=s2[j]; /* concat */
s1[i]='\0';
printf("Ur name is %s",s1);
getch();
}
```

The bottom window shows the execution of the program. It displays the prompts and the user's input: "Enter first name Kishore", "Enter last name Naidu", and the resulting output "Ur name is KishoreNaidu".

```
Enter first name Kishore
Enter last name Naidu
Ur name is KishoreNaidu
```

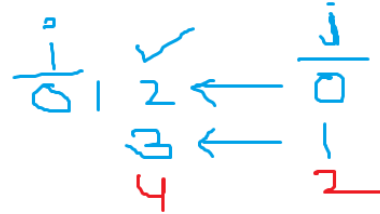


```
for( i=0; s1[i]!='\0'; i++) ; /* strlen */
```

```
for( j=0; s2[j]!='\0'; j++) s1[i]=s2[j];
```

```
s1[i] = '\0';
```

```
p(s1);
```



With space:

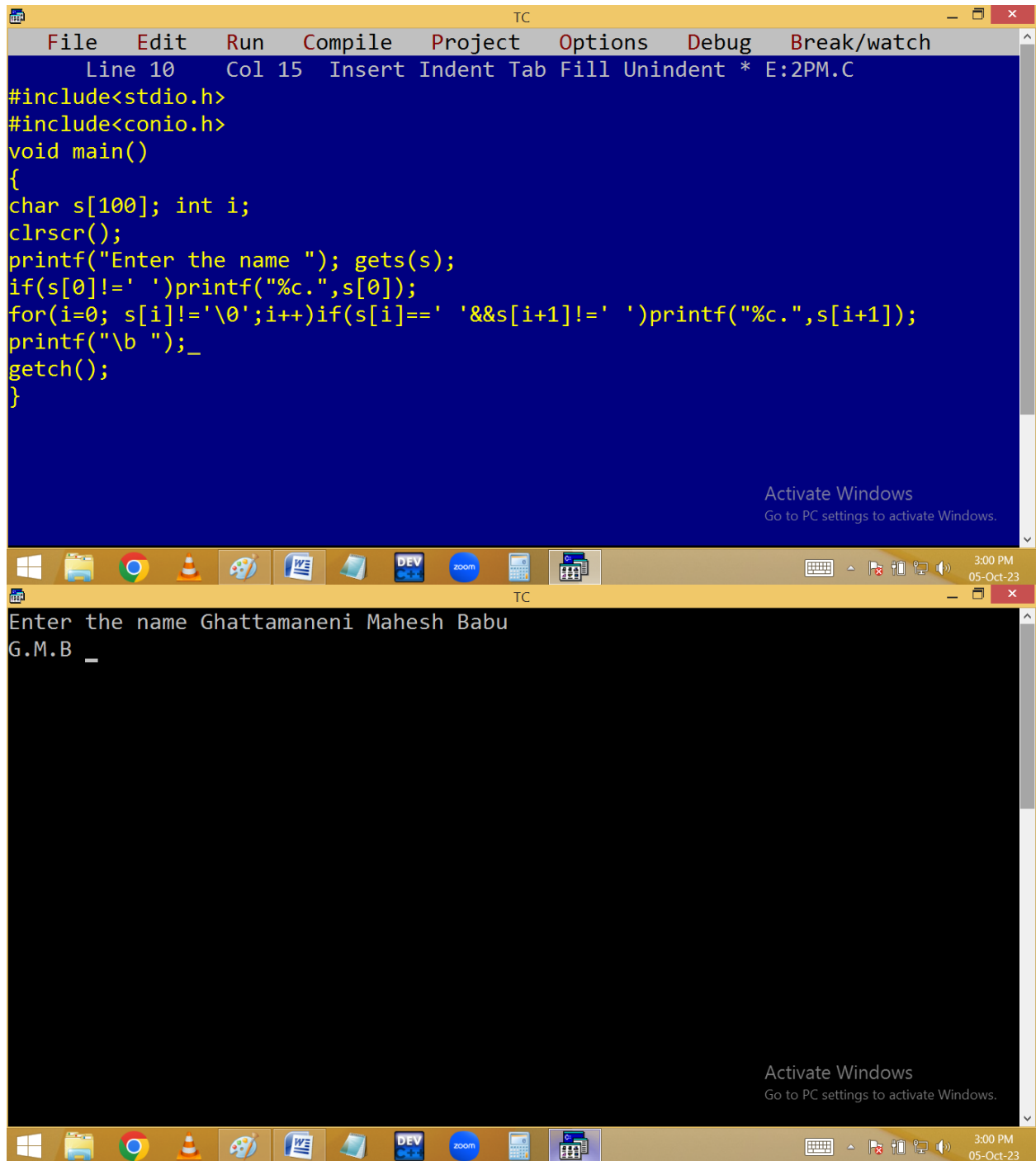
The image shows two windows of the Turbo C++ (TC) IDE. The top window displays the source code for a C program named E:2PM.C. The code includes headers for stdio.h and conio.h, and implements a main function that prompts the user for a first and last name, concatenates them with a space, and prints the result. The bottom window shows the program's execution output, where the user has entered 'Jhanvi' for the first name and 'kapoor' for the last name, resulting in the output 'Ur name is Jhanvi kapoor'.

```
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 18 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s1[100],s2[100]; int i,j;
clrscr();
printf("Enter first name "); gets(s1);
printf("Enter last name "); gets(s2);
for(i=0; s1[i]!='\0';i++); /* strlen */
s1[i++]=' ';
for( j=0; s2[j]!='\0';j++,i++)s1[i]=s2[j]; /* concat */
s1[i]='\0';
printf("Ur name is %s",s1);
getch();
}
```

Enter first name Jhanvi
Enter last name kapoor
Ur name is Jhanvi kapoor

Abbreviation:

Surya kumar yadav → S k y



The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code of a C program in a blue editor window. The code includes headers for `stdio.h` and `conio.h`, and implements a `main` function that reads a name and prints it character by character. The bottom screenshot shows the same IDE with the program executed, displaying the input name "Ghattamaneni Mahesh Babu" and the prompt "G.M.B _".

TC

File Edit Run Compile Project Options Debug Break/watch

Line 10 Col 15 Insert Indent Tab Fill Unindent * E:2PM.C

```
#include<stdio.h>
#include<conio.h>
void main()
{
char s[100]; int i;
clrscr();
printf("Enter the name "); gets(s);
if(s[0]!=' ')printf("%c.",s[0]);
for(i=0; s[i]!='\0';i++)if(s[i]!=' '){printf("%c.",s[i]);}
printf("\b ");_
getch();
}
```

Activate Windows
Go to PC settings to activate Windows.

3:00 PM
05-Oct-23

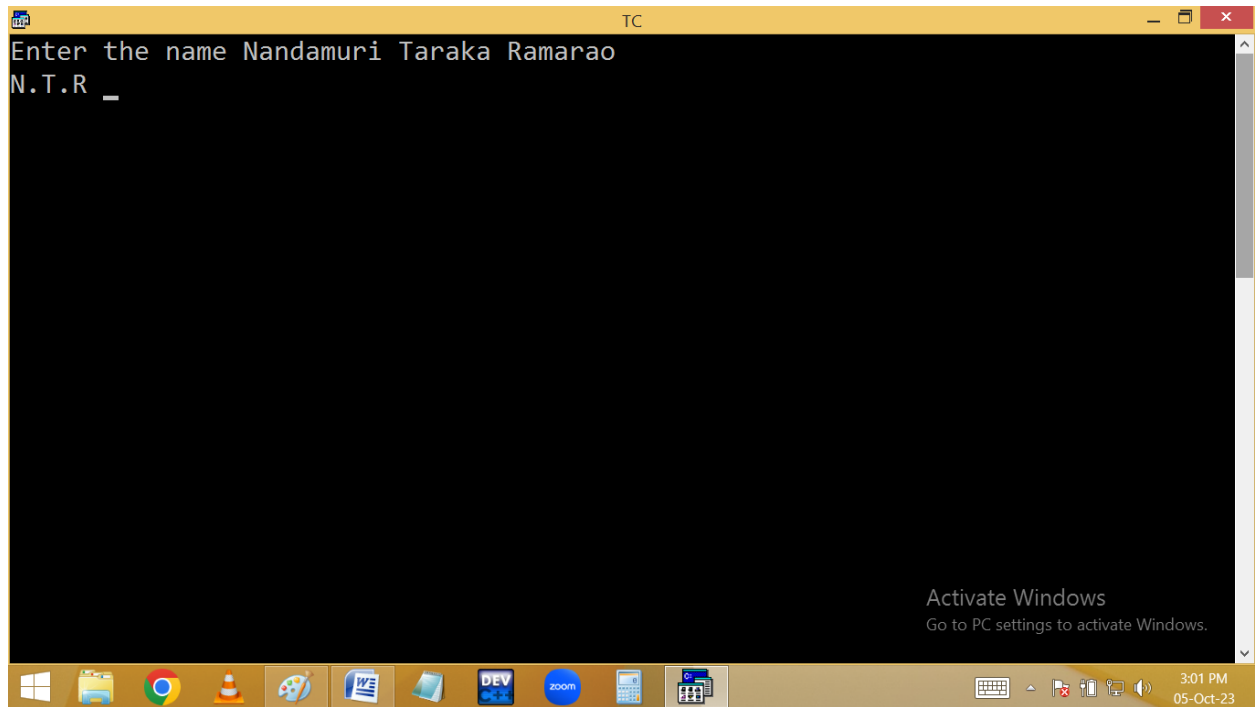
TC

Enter the name Ghattamaneni Mahesh Babu

G.M.B _

Activate Windows
Go to PC settings to activate Windows.

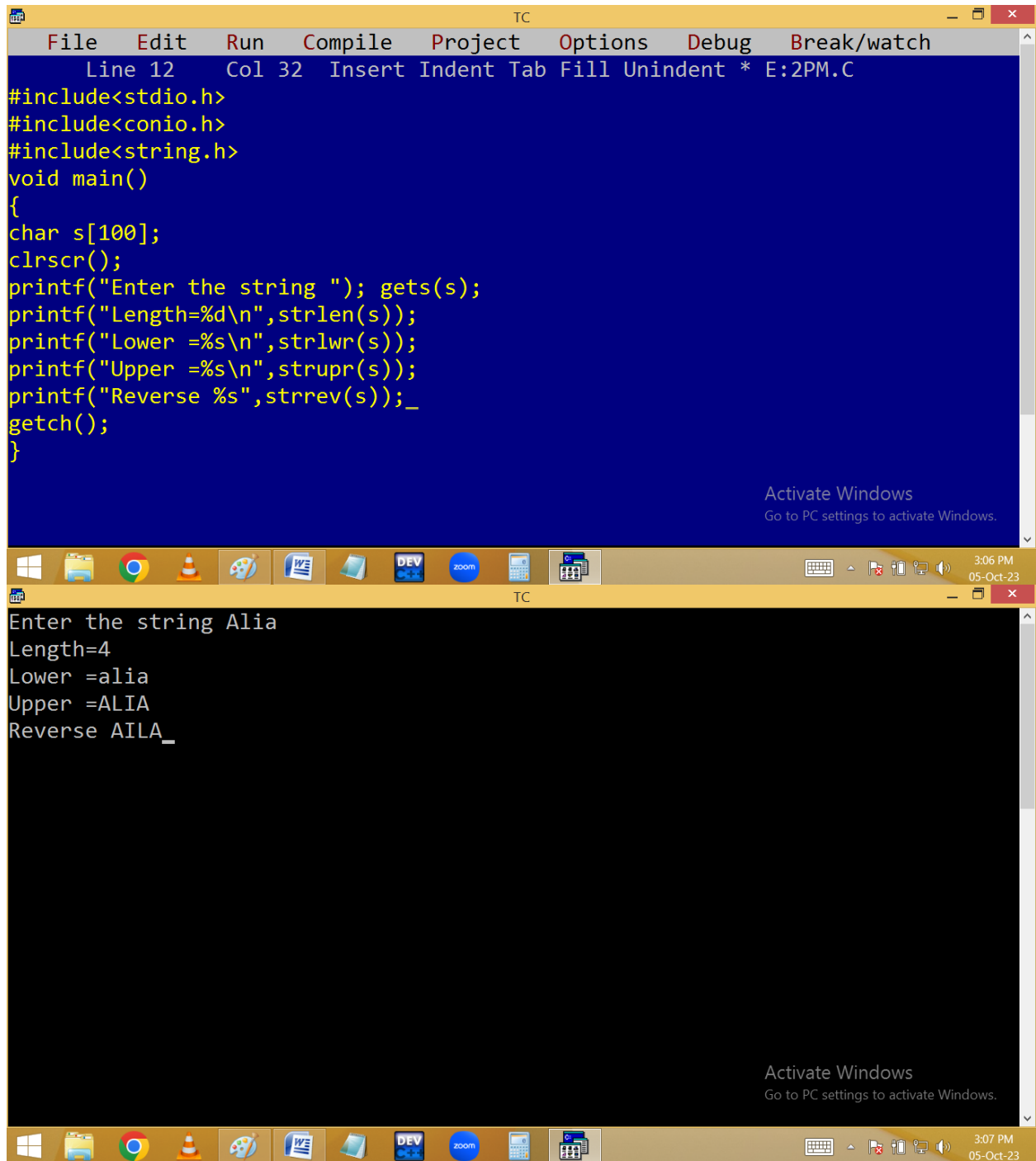
3:00 PM
05-Oct-23



String library functions:

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

1. **strlen()**: It return the string length.
2. **strrev()**: Return reverse string.
3. **strlwr()**: Converts into lower case.
4. **strupr()**: Converts into upper case.



The image shows two windows of the Turbo C++ (TC) IDE. The top window displays the source code for a C program named E:2PM.C. The code includes headers for stdio.h, conio.h, and string.h, and defines a main function that prompts the user for a string, calculates its length, converts it to lowercase and uppercase, and reverses it. The bottom window shows the program's execution output for the input 'Alia'.

```
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 32 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[100];
clrscr();
printf("Enter the string "); gets(s);
printf("Length=%d\n",strlen(s));
printf("Lower =%s\n",strlwr(s));
printf("Upper =%s\n",strupr(s));
printf("Reverse %s",strrev(s));_
getch();
}
```

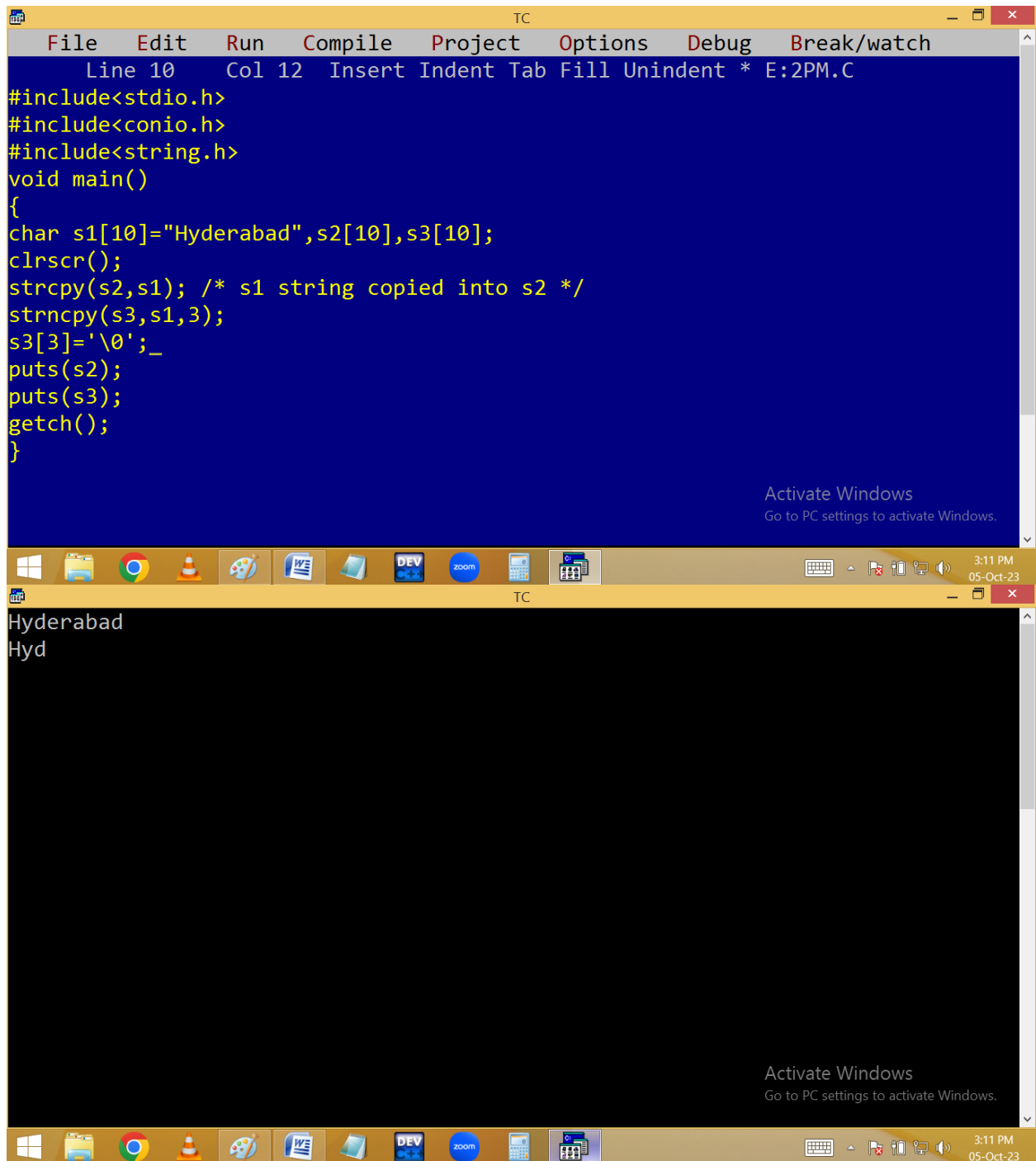
Enter the string Alia
Length=4
Lower =alia
Upper =ALIA
Reverse AILA_

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

Strcpy(destination string, source string);

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

strcpy(dest string, source string, no of char's);



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 12 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[10]="Hyderabad",s2[10],s3[10];
clrscr();
strcpy(s2,s1); /* s1 string copied into s2 */
strncpy(s3,s1,3);
s3[3]='\0';_
puts(s2);
puts(s3);
getch();
}
```

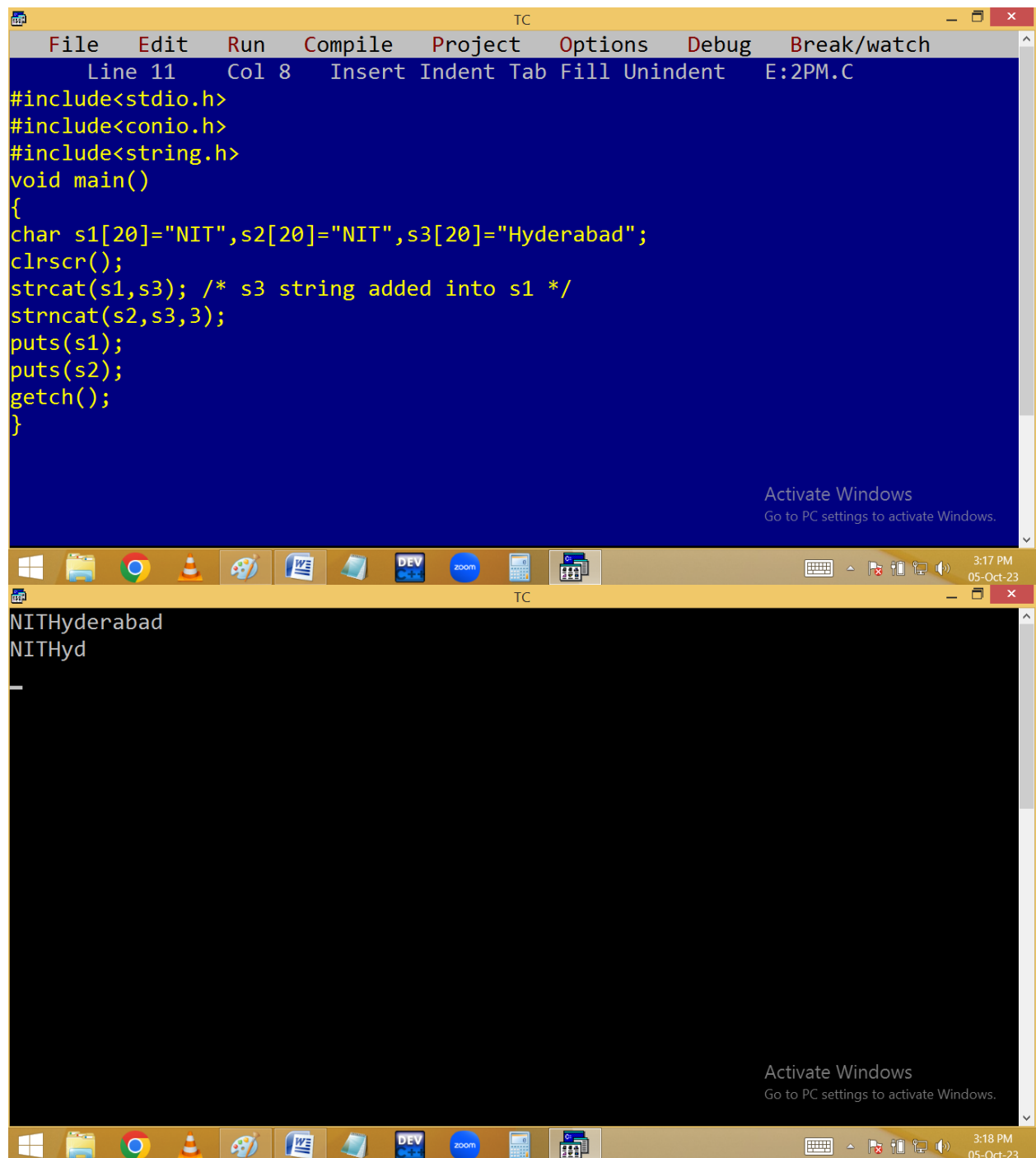
Hyderabad
Hyd

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

strcat(string1, string2);

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

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



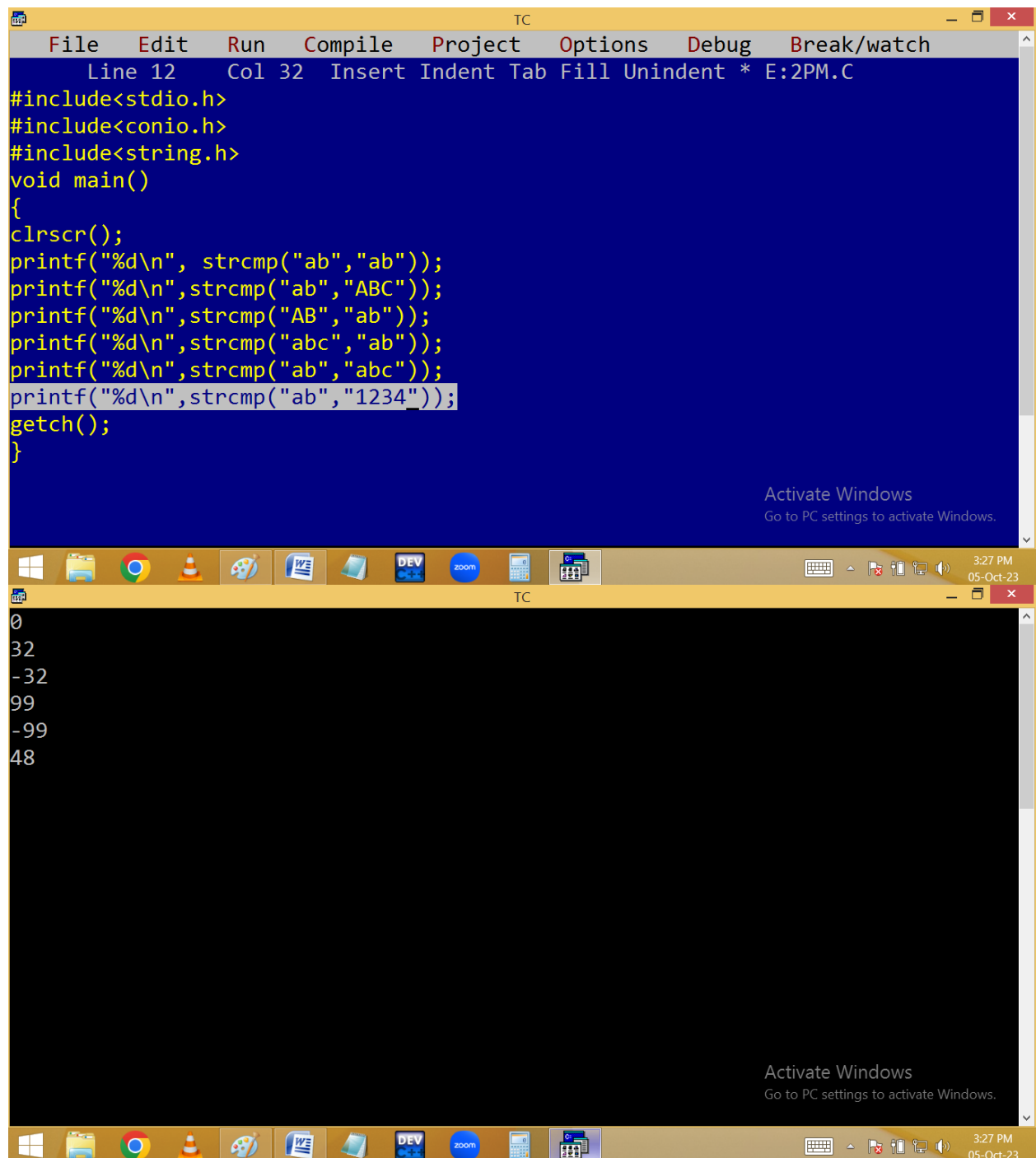
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 8 Insert Indent Tab Fill Unindent E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s1[20]="NIT",s2[20]="NIT",s3[20]="Hyderabad";
clrscr();
strcat(s1,s3); /* s3 string added into s1 */
strncat(s2,s3,3);
puts(s1);
puts(s2);
getch();
}
```

NITHyderabad
NITHyd

Activate Windows
Go to PC settings to activate Windows.

9. **strcmp()**: It compare two string based on **ASCII** values and when 1st difference found, it returns the difference value.

Strcmp(string1, string2);



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 32 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
clrscr();
printf("%d\n", strcmp("ab","ab"));
printf("%d\n",strcmp("ab","ABC"));
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 compare two strings by ignoring case. i.e. in stricmp() lower and upper are same. If matching char not found or different data type available in 2nd string, the first string char taken in upper case.

Stricmp(string1, string2);

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

```
#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, which consists of seven lines of integers: 0, 0, 0, 67, -99, 16, and 16. The taskbar at the bottom includes icons for Windows, File Explorer, Google Chrome, VLC media player, MS Paint, Word, Notepad, DEV C++, Zoom, and a calculator. The system clock in the bottom right corner indicates 3:33 PM on 05-Oct-23. An 'Activate Windows' watermark is visible in the bottom right of both windows.

11. **strstr():** It searches the sub string availability in main string and if found, it returns the sub string address. If sub string not found, it returns null.

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 13 Col 42 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char s[]="abcdef";
clrscr();
printf("%u\n", s);
printf("c addr = %u\n", strstr(s, "c"));
printf("c addr = %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();
}
```

The bottom window shows the output of the program:

```
65496
c addr = 65498
c addr = 0
cdef
(null)
c is 3 char
```

Both windows include a taskbar at the bottom with various application icons and a system tray showing the time as 3:41 PM on 05-Oct-23. An "Activate Windows" watermark is visible in the bottom right of each window.