3-dimensional arrays:

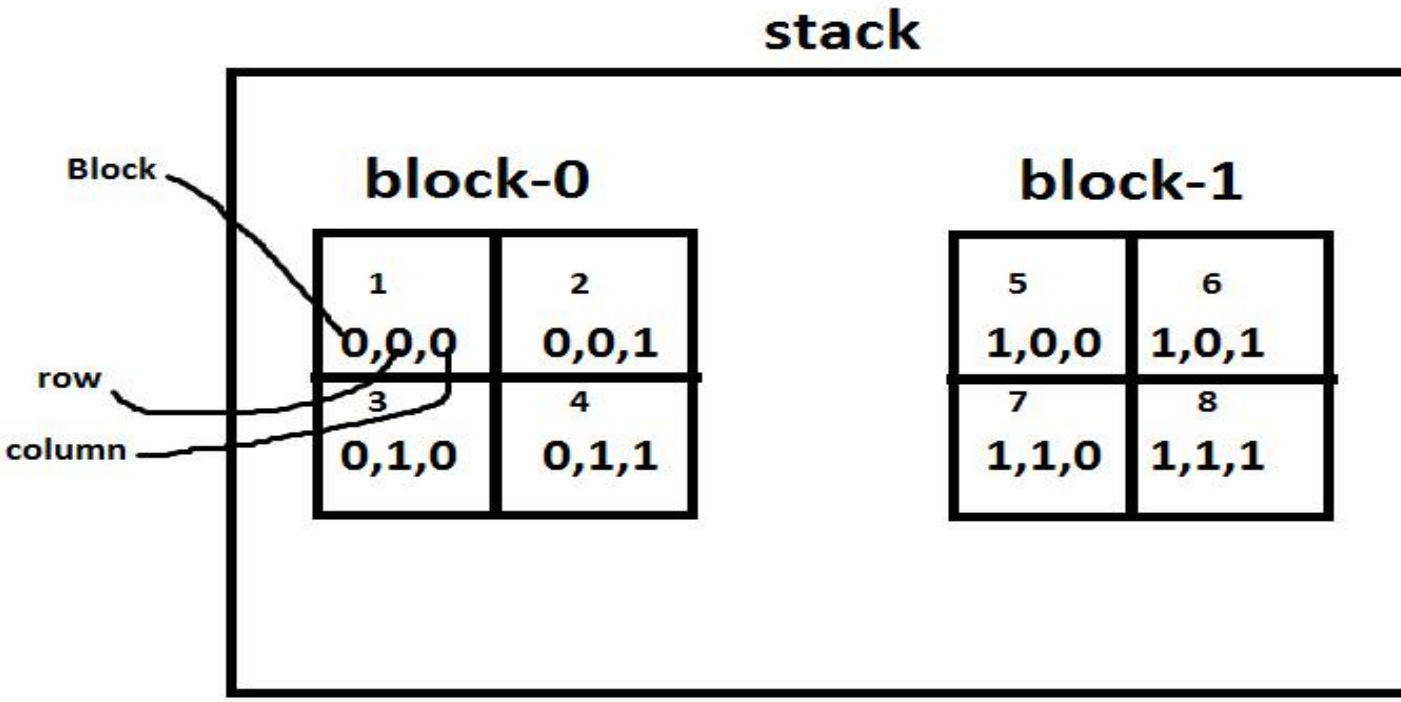
An array with several blocks, rows and columns.

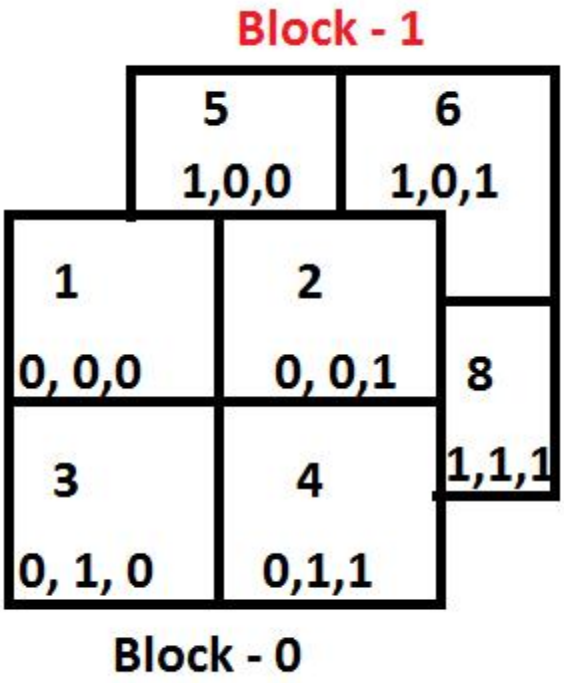
An array with 3 subscripting operators [ ] [ ] [ ].

Syntax:

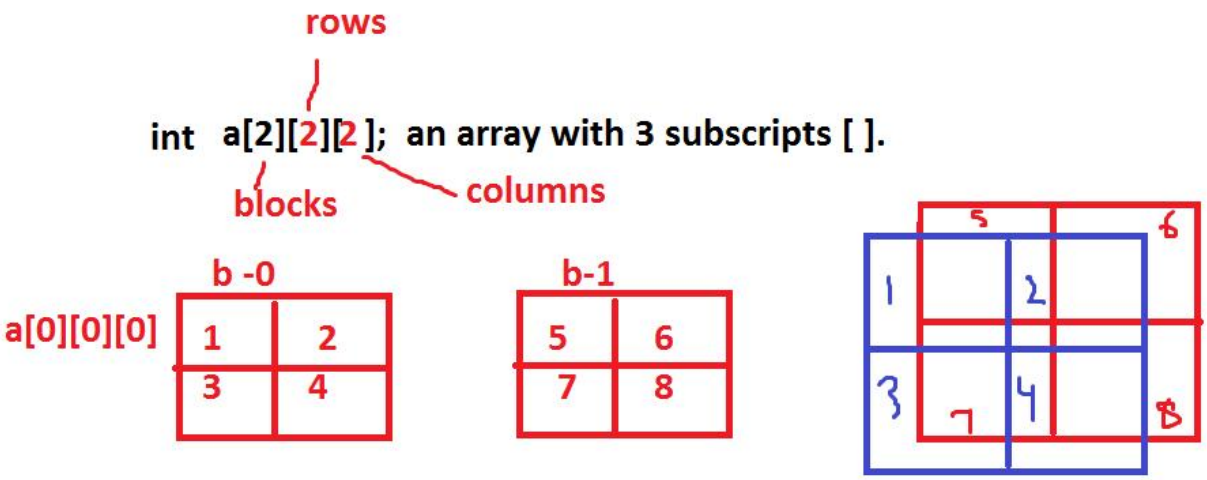
datatype variable [blocks] [rows] [columns];

Eg: int a[2][2][2]={1,2,3,4,5,6,7,8};





Eg:



eg: int class[2][60][6];

datatype class[sections][stus][marks];

TC

File Edit Run Compile Project Options Debug Break/

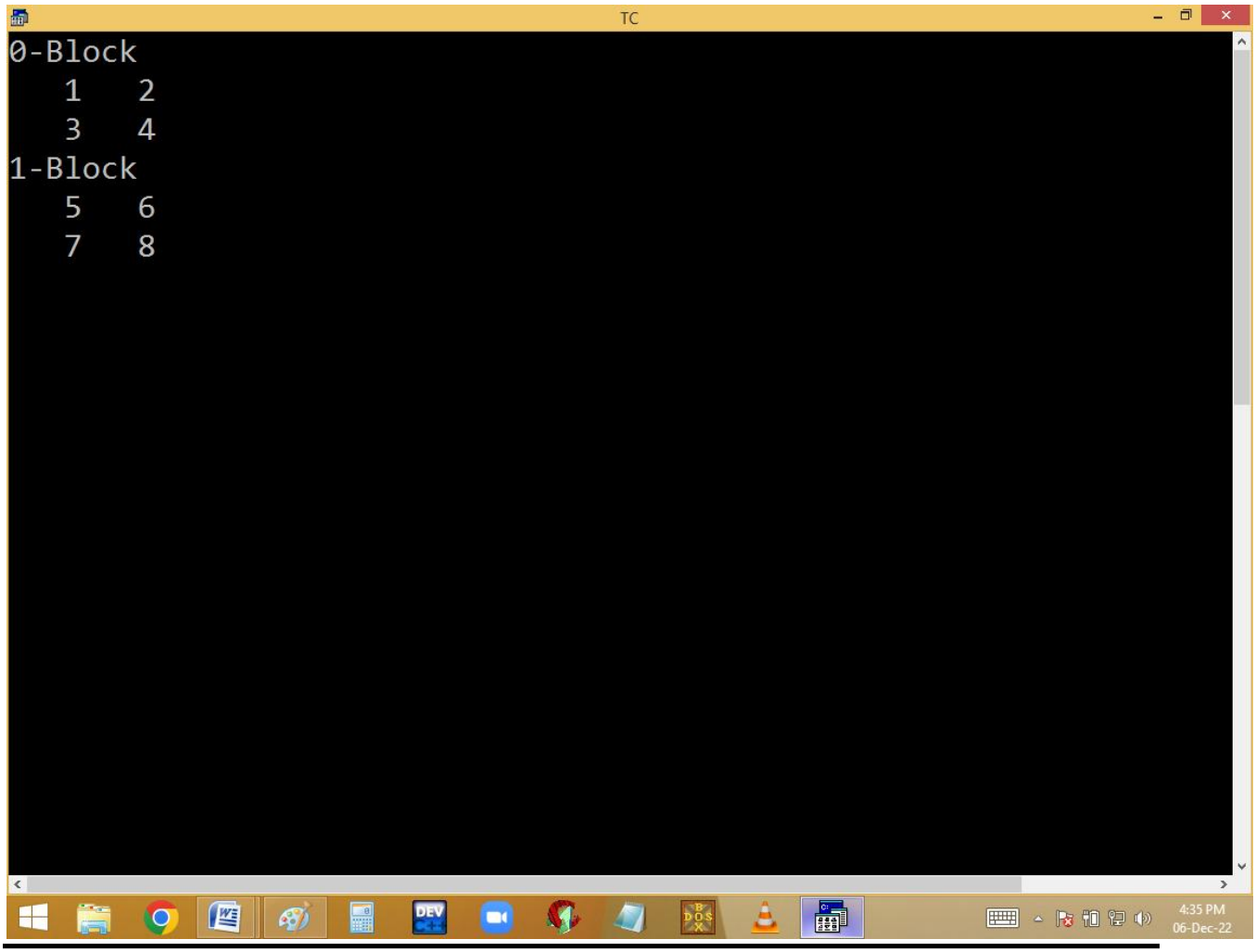
Edit

Line 1 Col 2 Insert Indent Tab Fill Unindent * E:NONAM

```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][2][2]={1,2,3,4,5,6,7,8},b,r,c;
clrscr();
for(b=0;b<2;b++)
{
printf("%d-Block\n",b);
for(r=0;r<2;r++)
{
for(c=0;c<2;c++) printf("%4d",a[b][r][c]);
printf("\n");
}
}
getch();
}
```

Watch

4:35 PM
06-Dec-22



4-dimensional array:

An array with several sets, blocks, rows and columns.

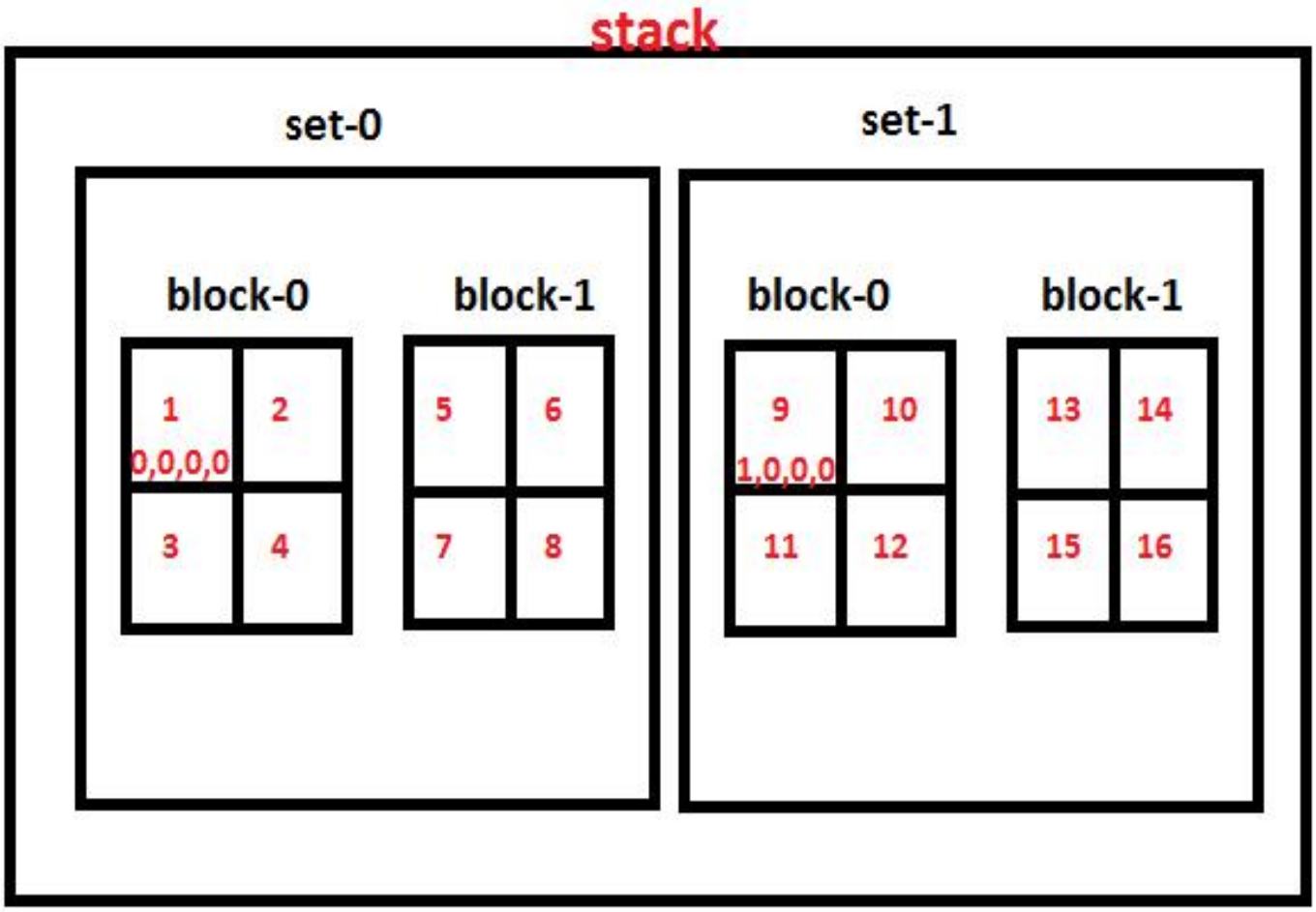
An array with 4 subscripting operators [ ] [ ] [ ] [ ] .

Syntax:

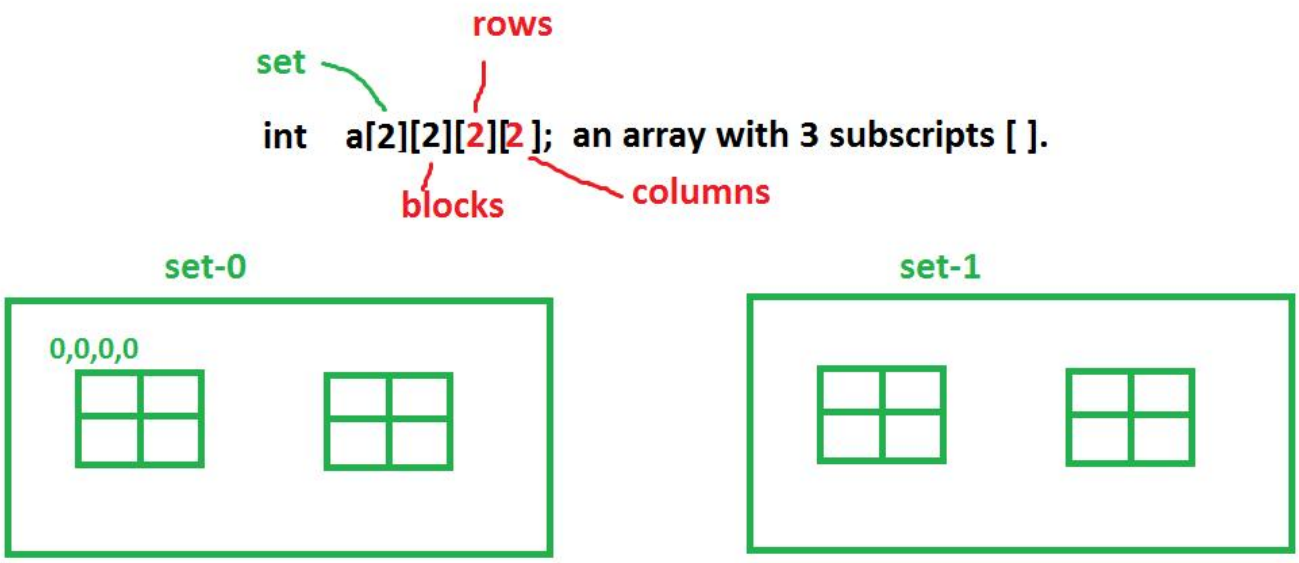
datatype variable [sets] [blocks] [rows] [cols];

eg:

```
int a[2] [2] [2] [2]= {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16};
```



```
datatype   var[class][sec][stu][marks];
int        school[5][2][60][6];
```



STRINGS

- A group of characters is called string.
- It is one dimensional character array.
- It is alpha-numeric.
- It is an implicit pointer.
- It is a derived data type.

Note:

- One byte should be left for Null char(\0). Otherwise we are getting garbage or junk values. Null char indicates string is completed.
- String variable Size can't be less than string. Otherwise we are getting error.
- Using = operator, we can't copy a string into another. We have to use strcpy() or copy character by character manually.
- Using == (comparison) operator, we can't compare two strings. Use strcmp() or compare the characters one by one manually.

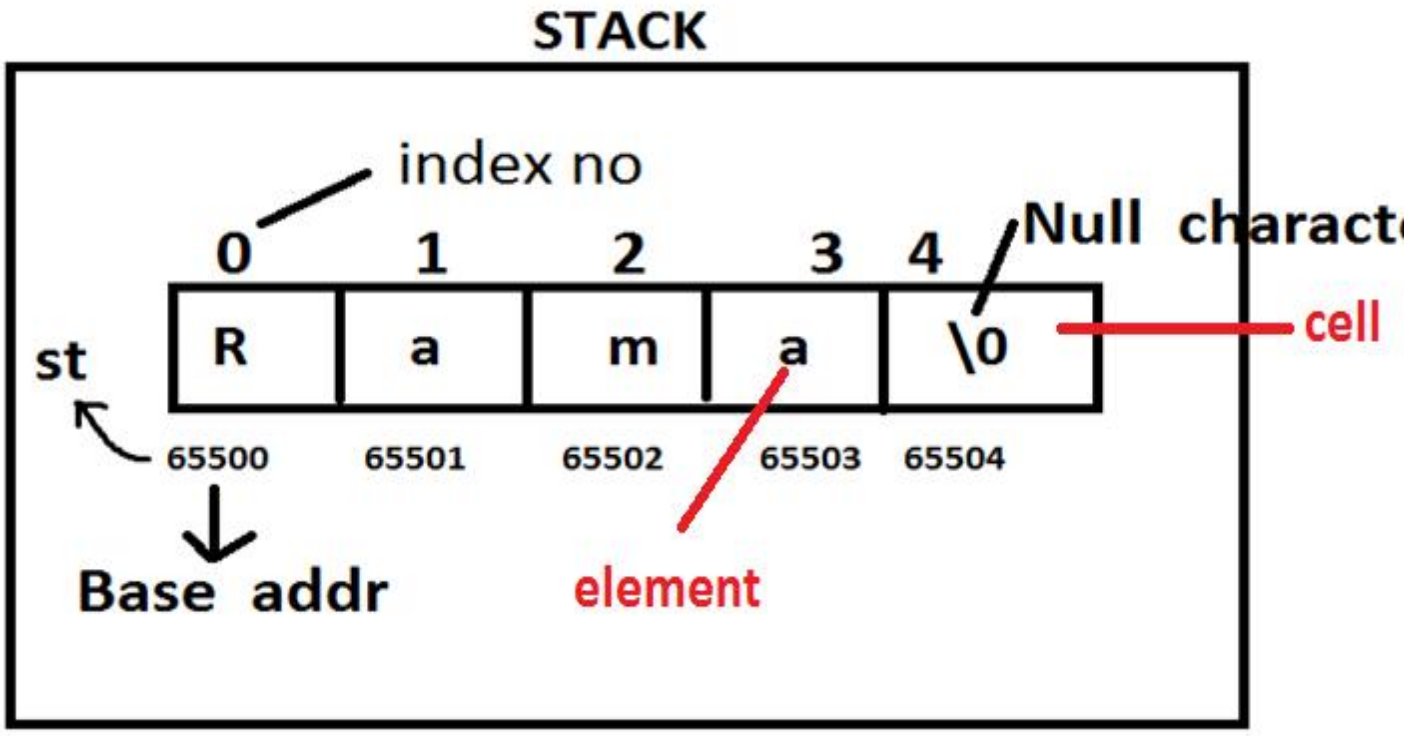
Syntax:

char variable [size] = "string";

or

char variable[ ]="string";

Eg: char st[5] = "Rama";



Note: String is implicit pointer because of string variable stores base address.

String declaration methods:

char st [5] =”rama”; Ok

char st [20] =” Naresh It”; Ok

char st [4] = { ‘r’, ‘a’, ‘m’ }; Ok → char array.

char st[3]= “ram”; It gives garbage values in printing.

char st [3] = “rama”; error

char st[0]; error

char st[0]=”abc”; Ok

char st[-5]; error

char st[5.5]; error

char st[5%3]; Ok ➔ char st[2];

char st[3+2]; ➔ st[5] ➔ Ok

char st[ ] =”Ram”; Ok.

char st[ ] ; error

int n=20;

char st[n]; No

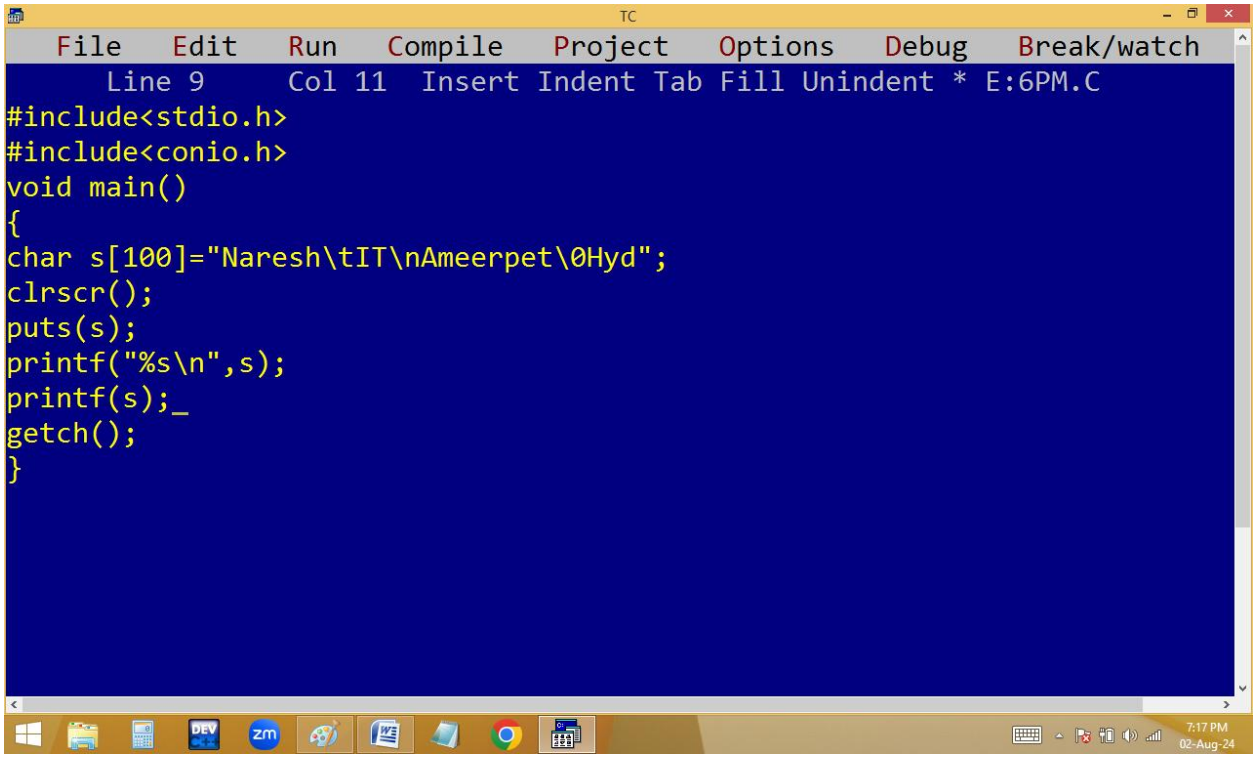
#define n 20

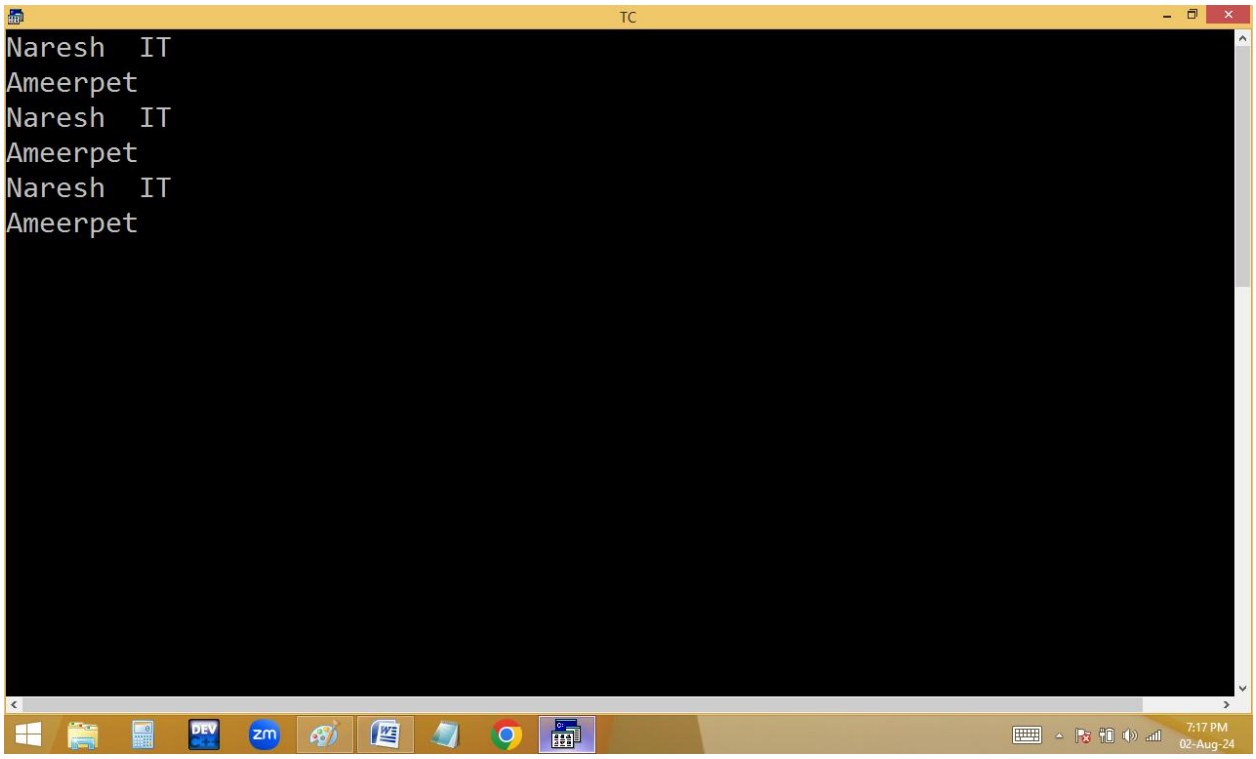
char st[n]; Ok

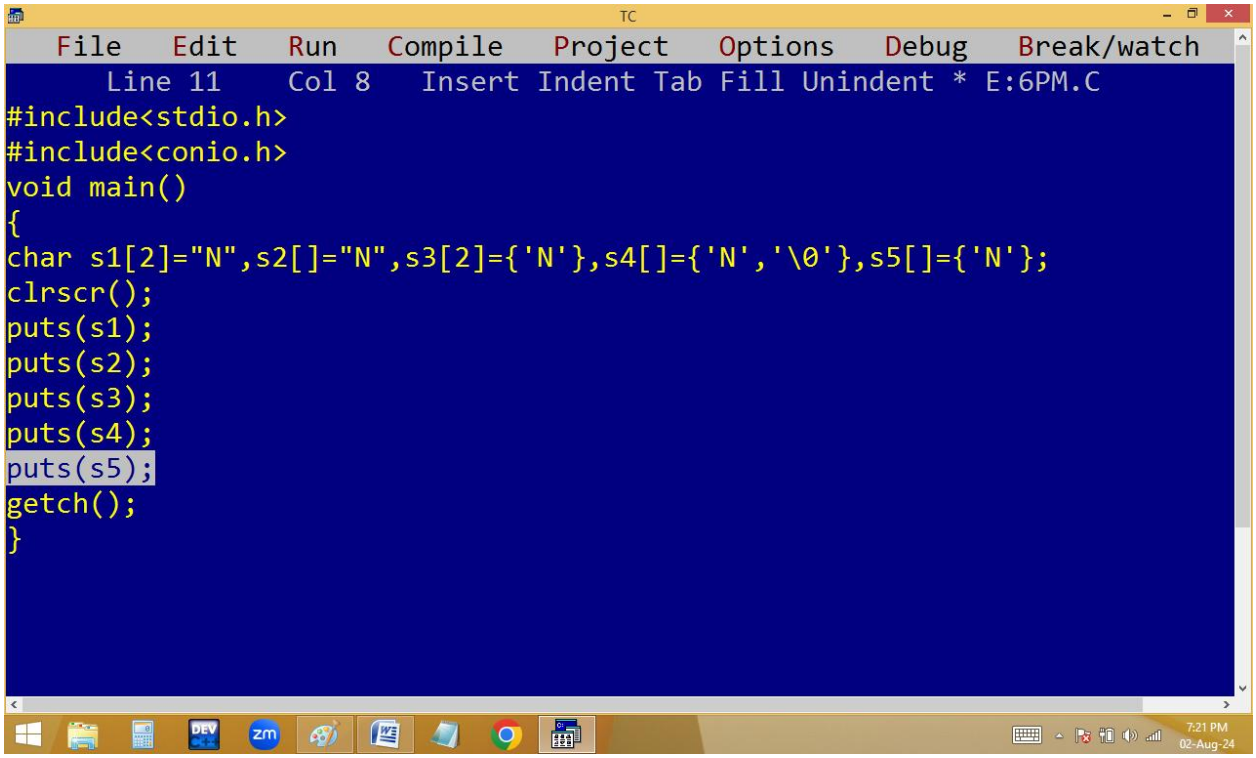
Note: String variable size always constant positive integer value.

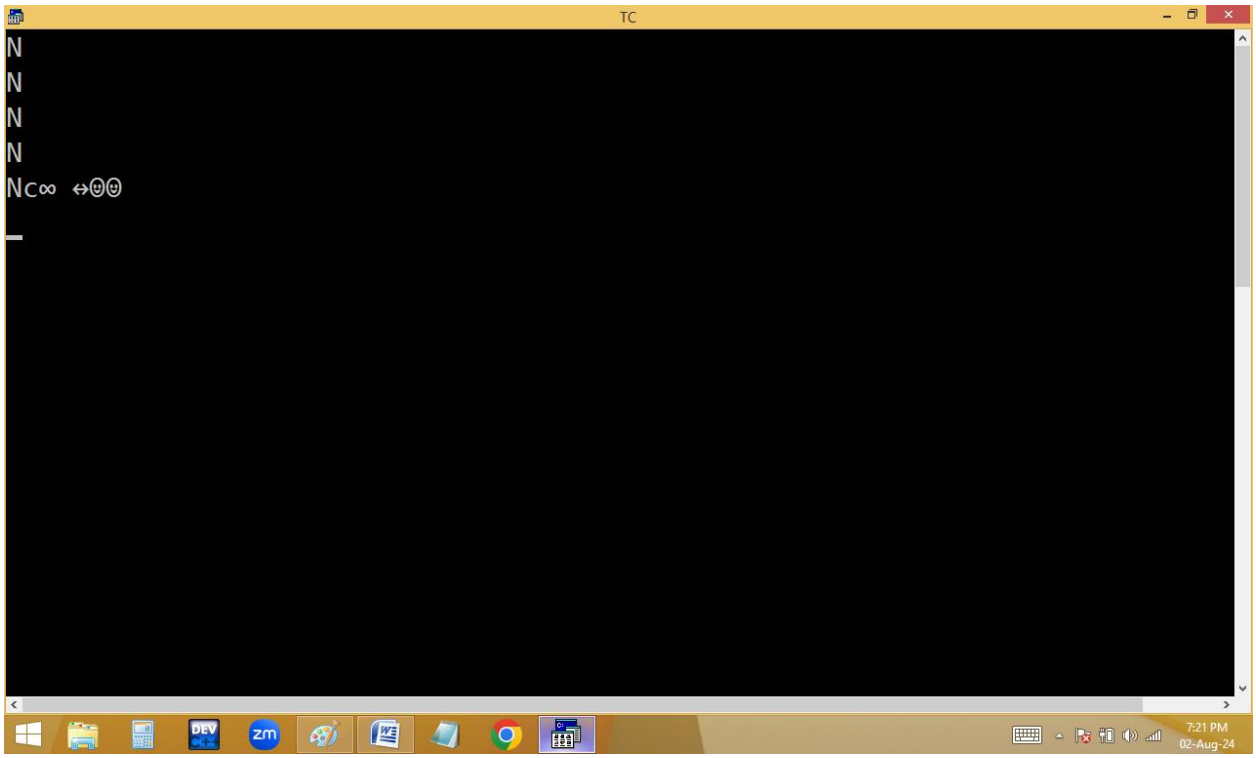
Eg:

Direct initialization of a string:

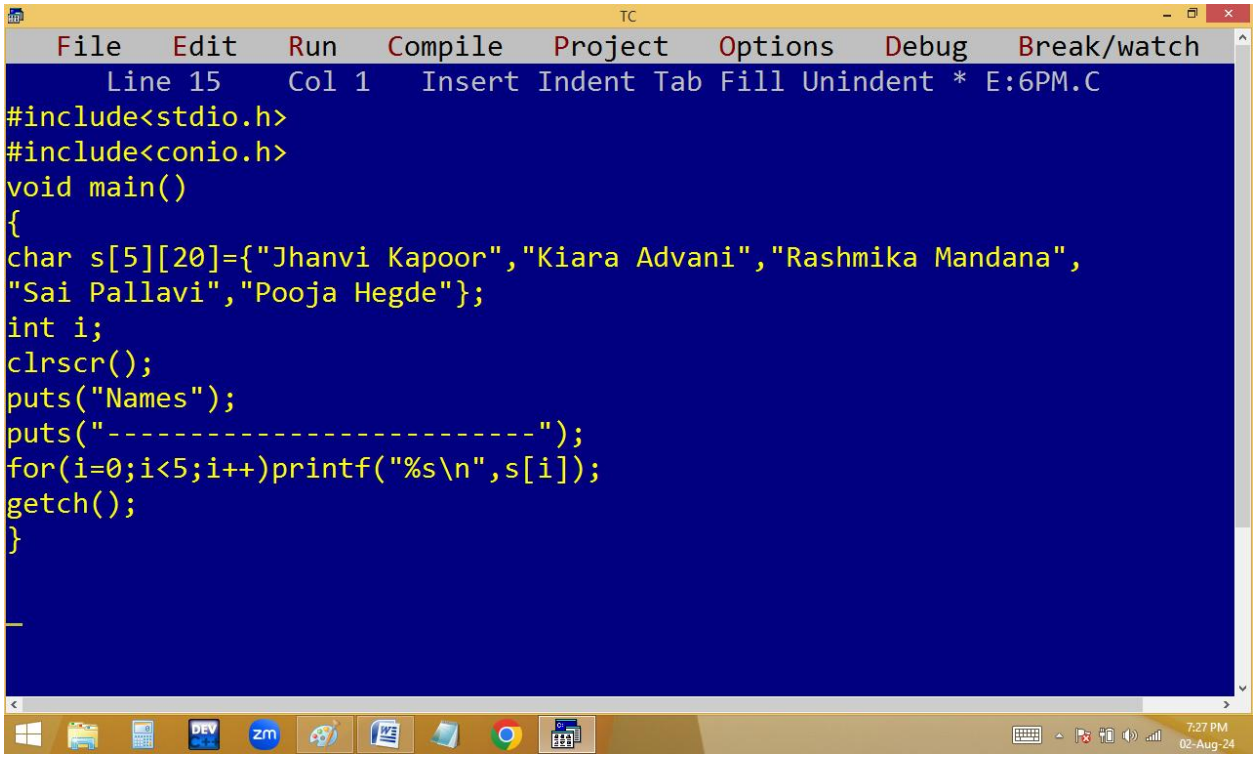

TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 11 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s[100]="Naresh\tIT\nAmeerpet\0Hyd";
clrscr();
puts(s);
printf("%s\n",s);
printf(s);_
getch();
}

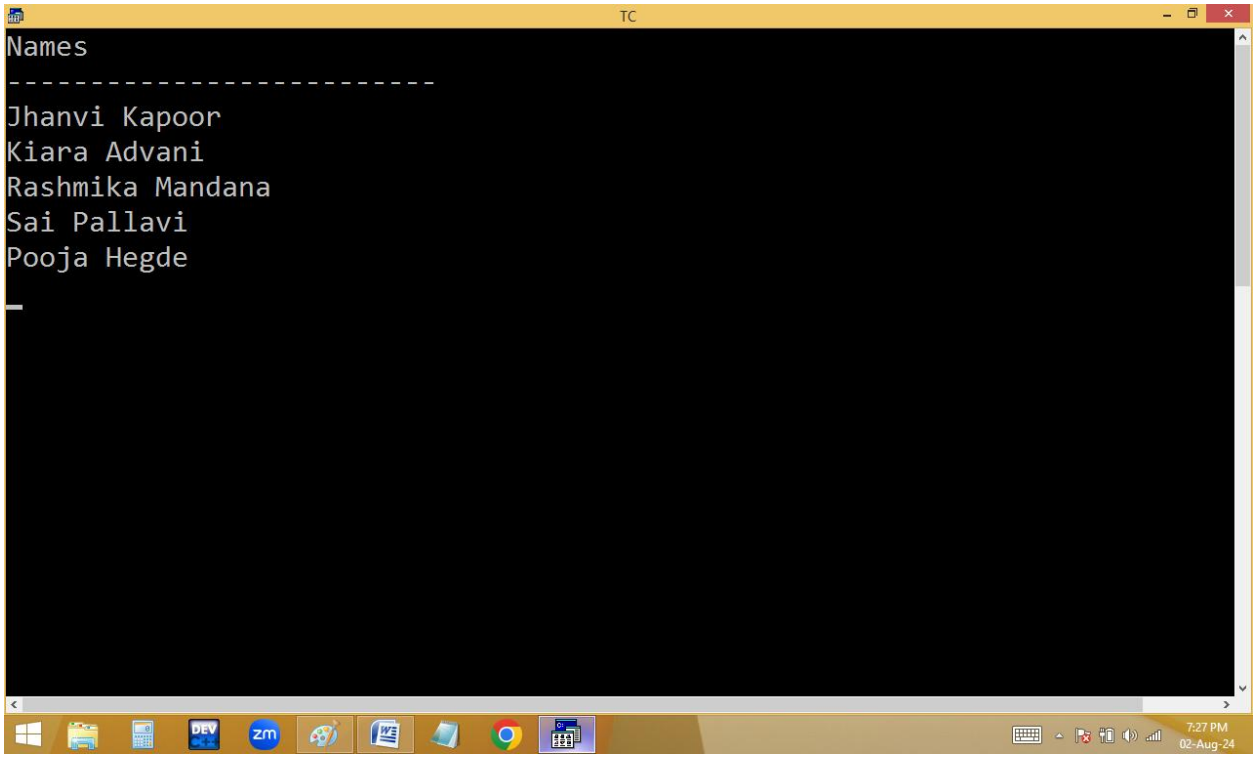

TC
Naresh IT
Ameerpet
Naresh IT
Ameerpet
Naresh IT
Ameerpet

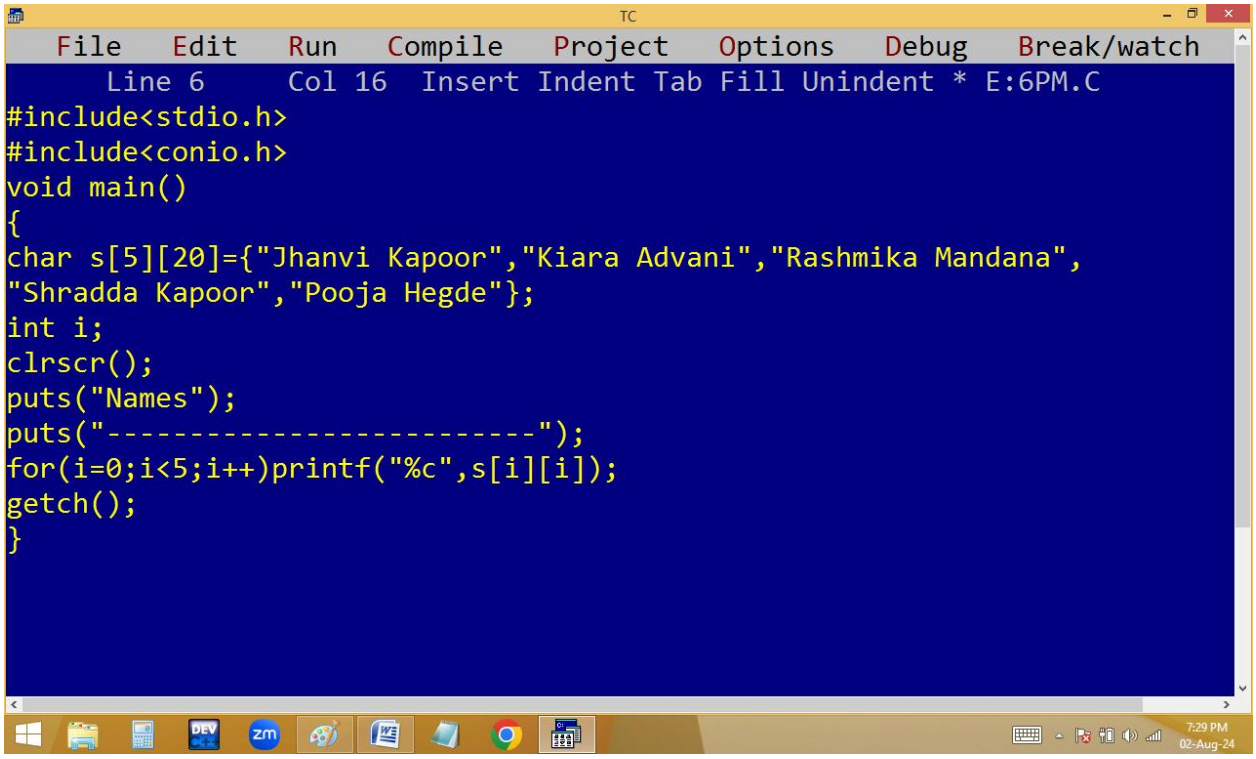

TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 8 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s1[2]="N",s2[]="N",s3[2]={'N'},s4[]={'N','\0'},s5[]={'N'};
clrscr();
puts(s1);
puts(s2);
puts(s3);
puts(s4);
puts(s5);
getch();
}

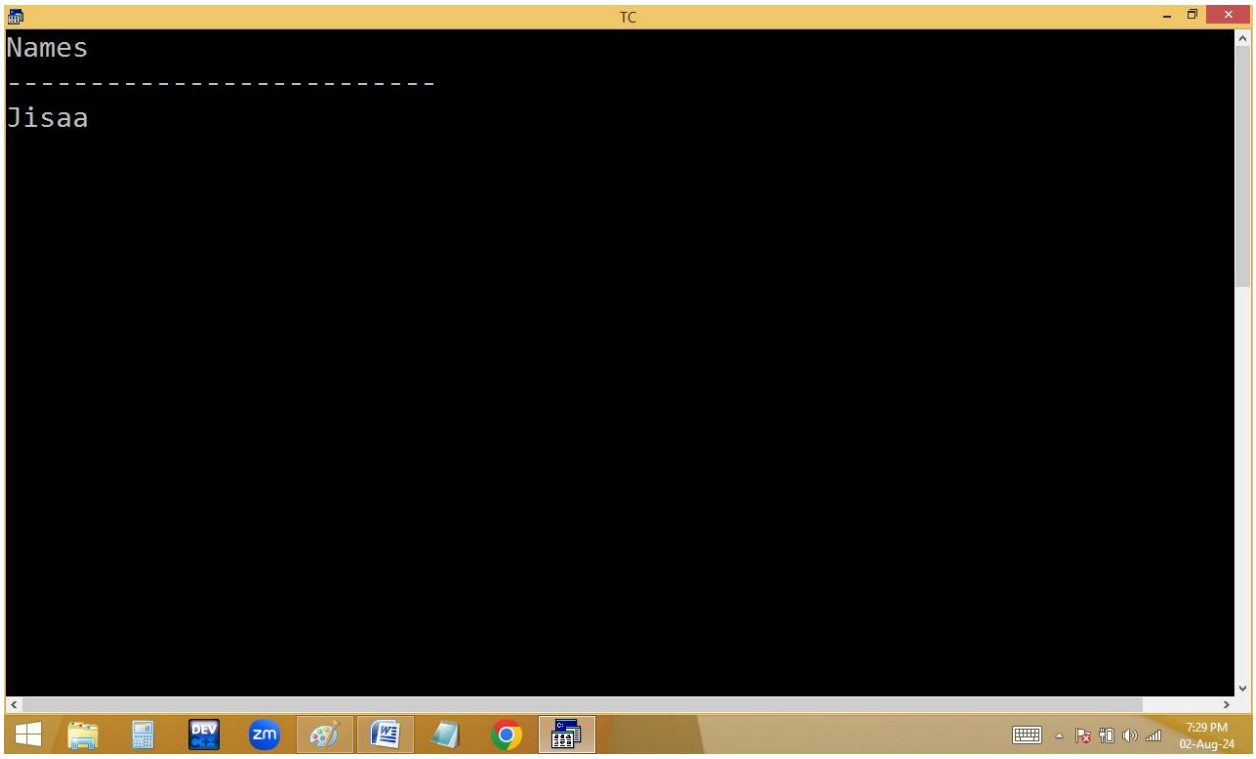

TC
N
N
N
N
Nc∞ ↔☺☺

Storing of multiple stirngs:



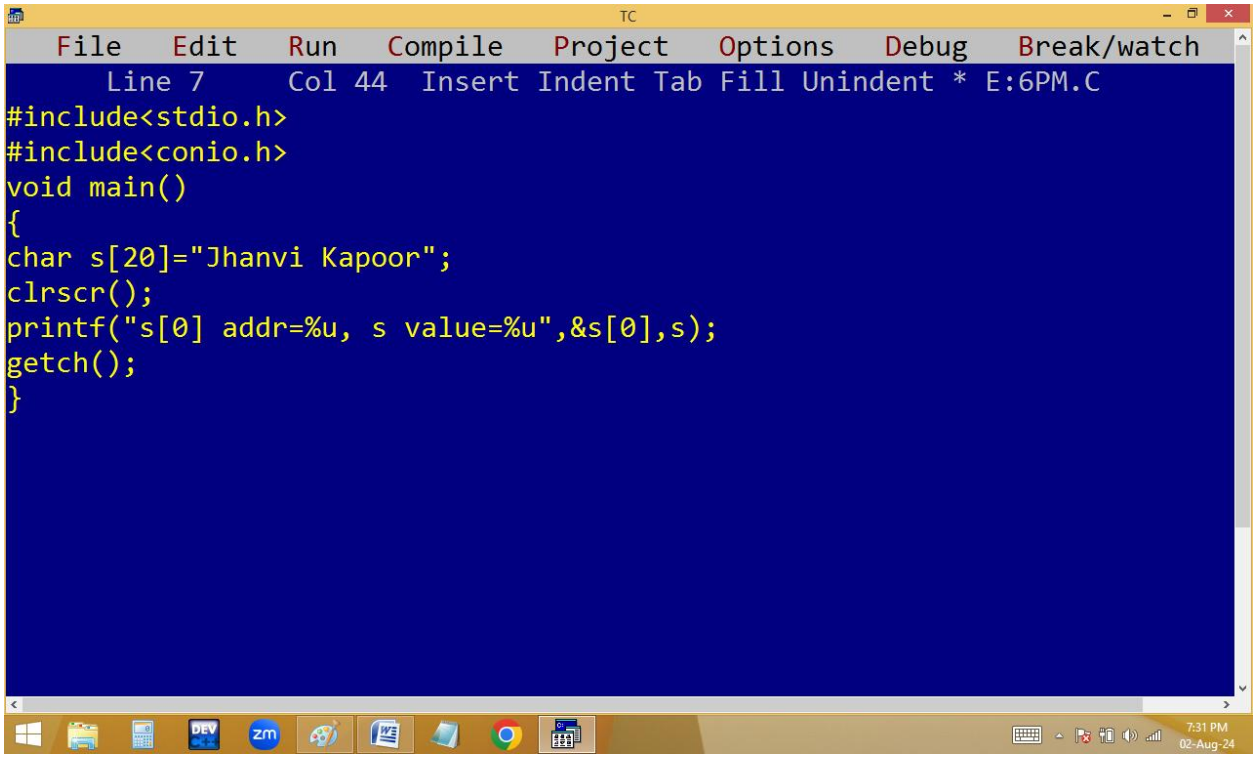


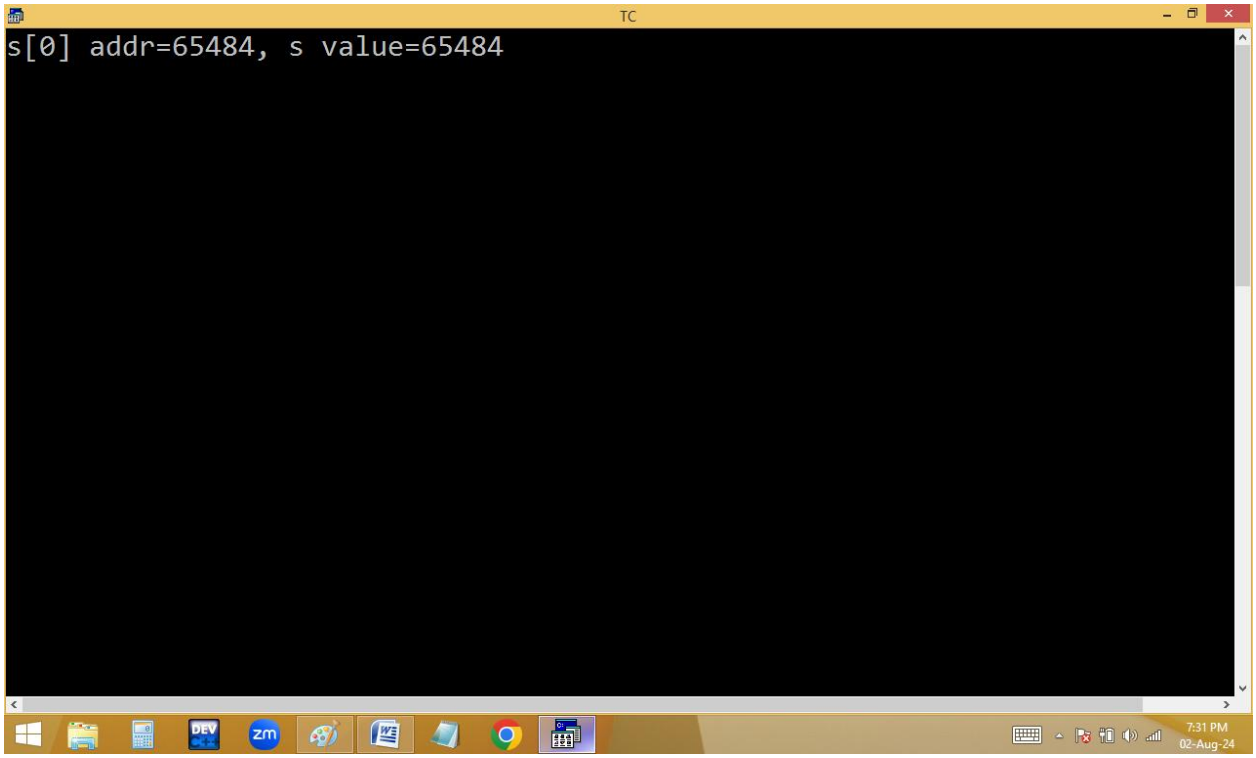

TC
File Edit Run Compile Project Options Debug Break/watch
Line 6 Col 16 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s[5][20]={"Jhanvi Kapoor","Kiara Advani","Rashmika Mandana",
"Shradda Kapoor","Pooja Hegde"};
int i;
clrscr();
puts("Names");
puts("--------------------------");
for(i=0;i<5;i++)printf("%c",s[i][i]);
getch();
}


TC
Names

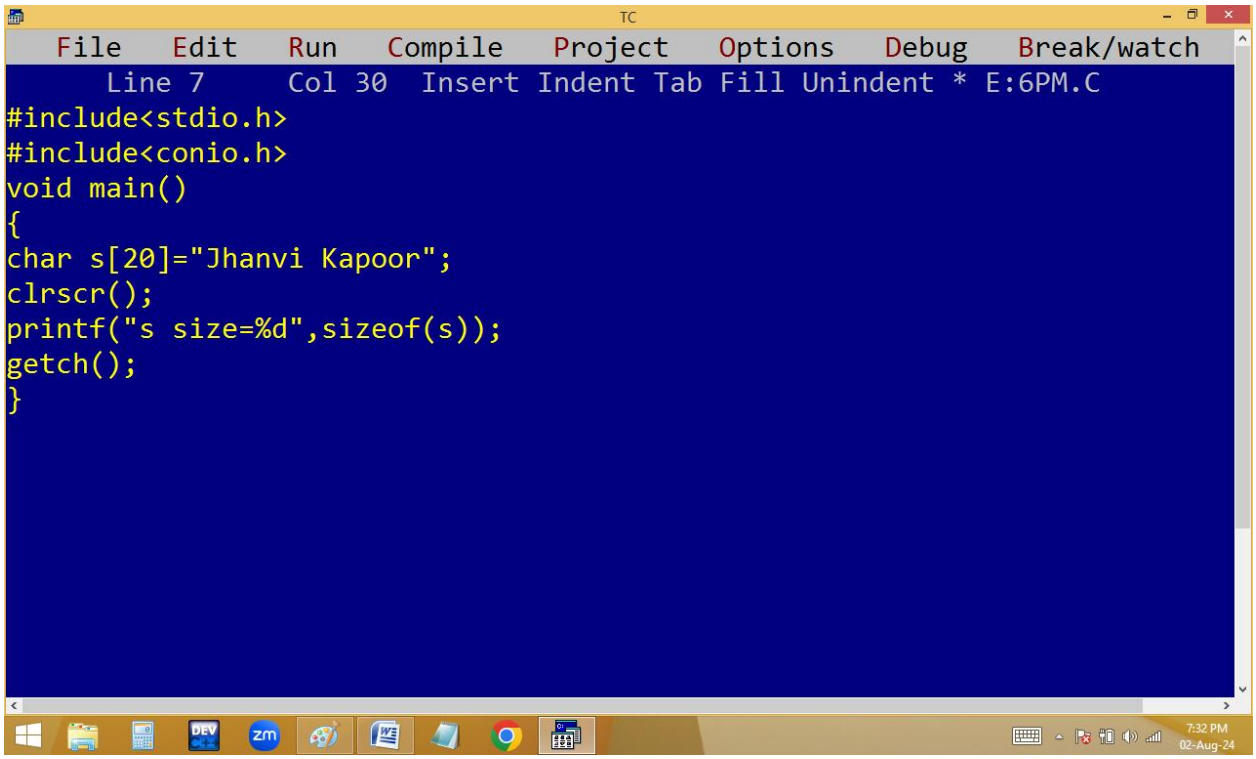
Jisaa

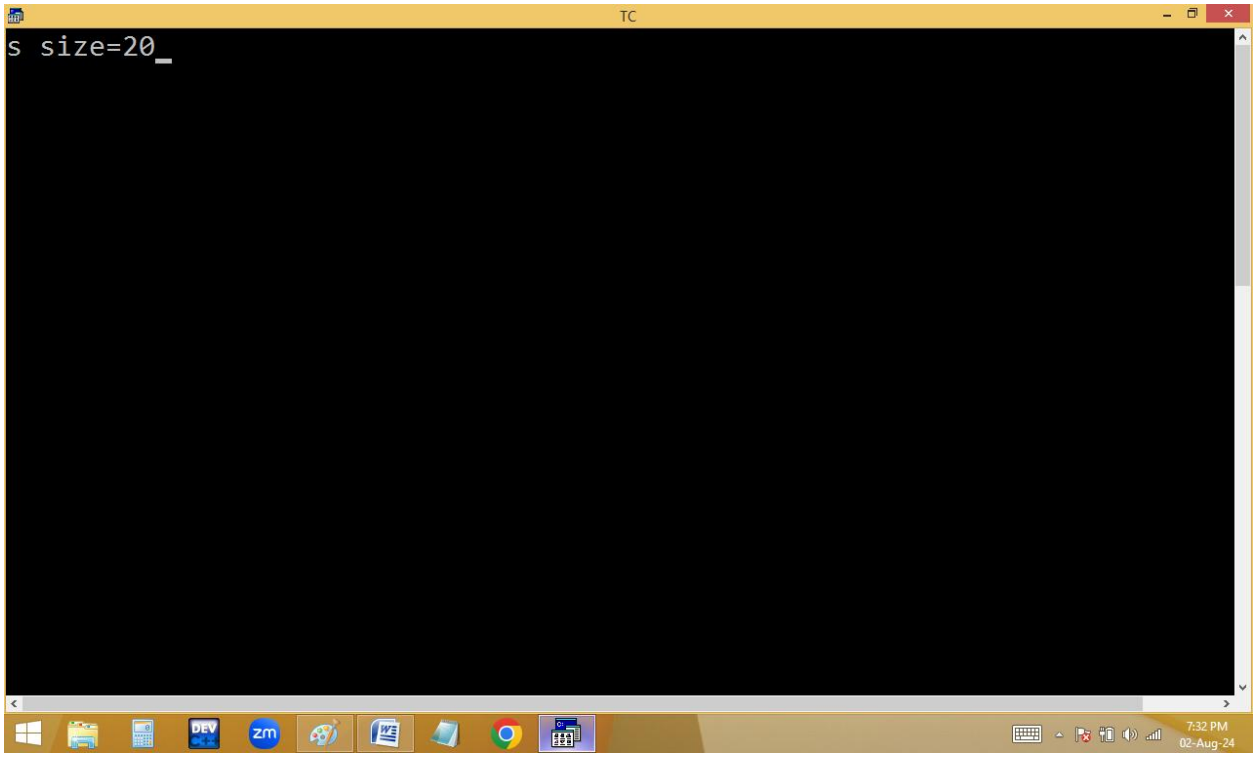
Finding string address:

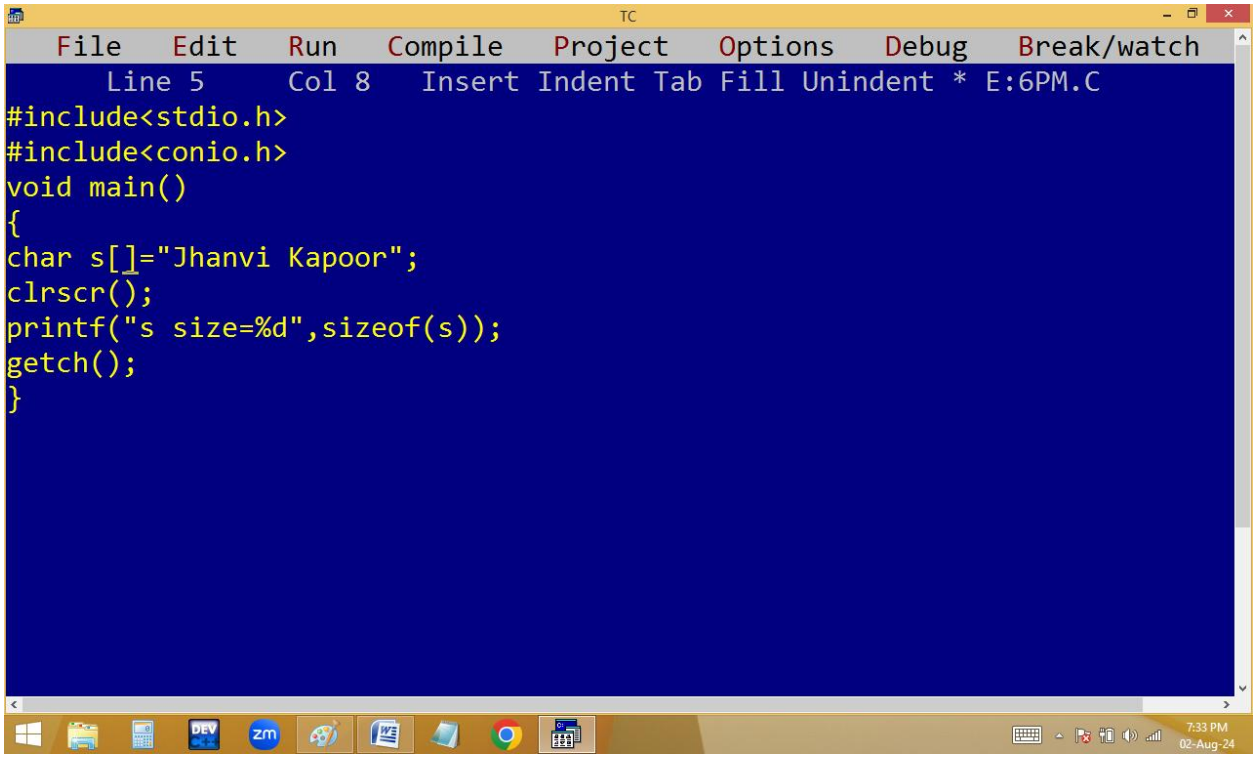


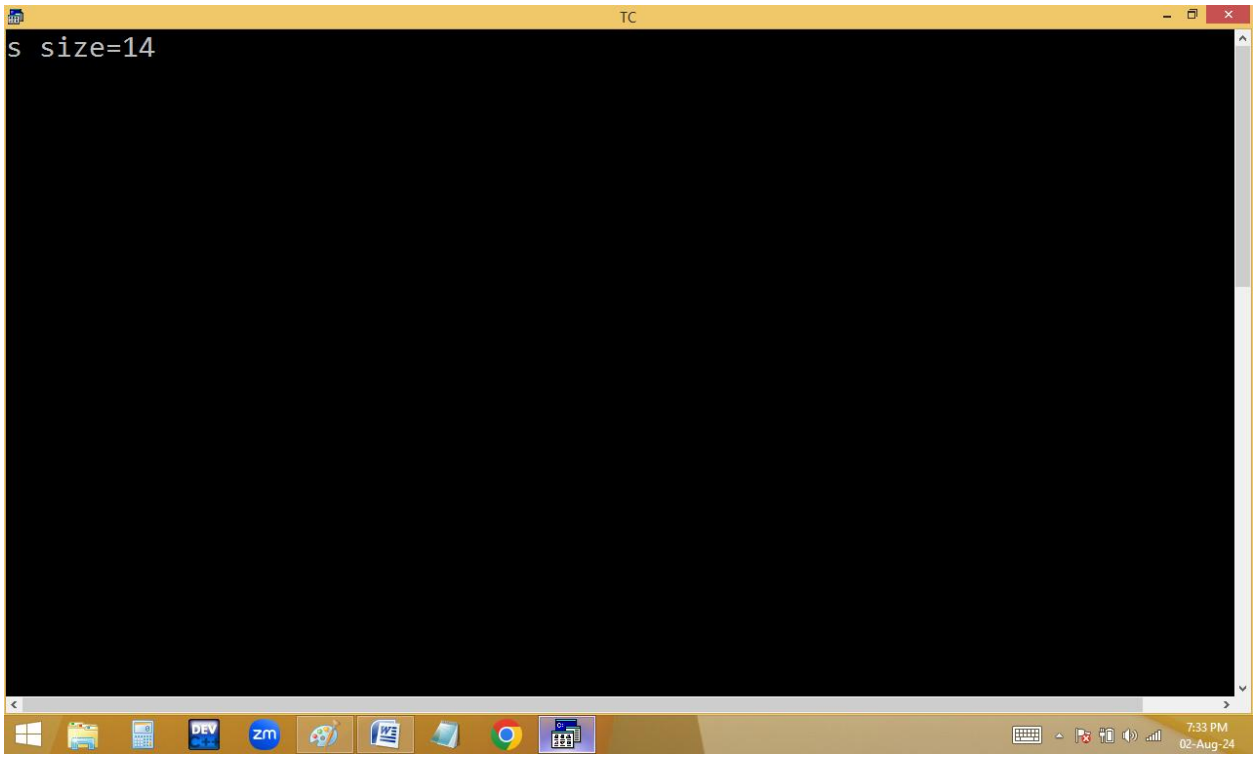


Finding string size:

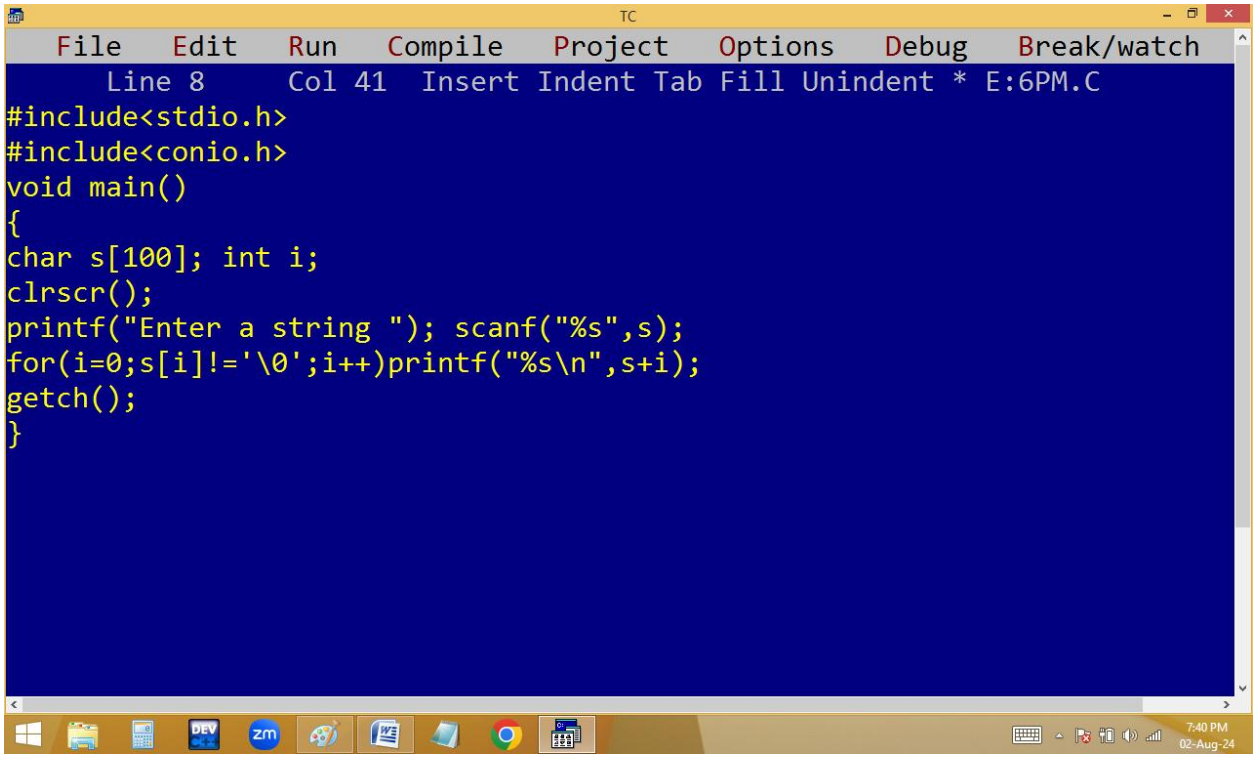


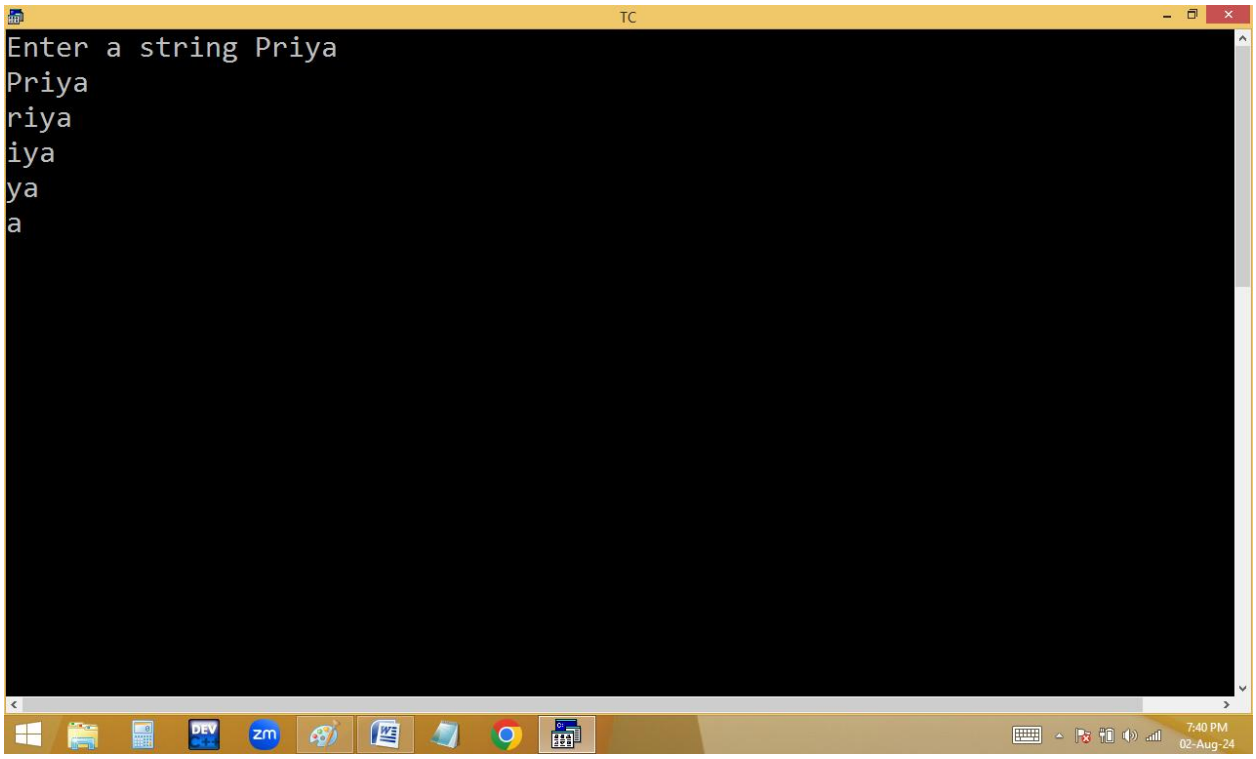


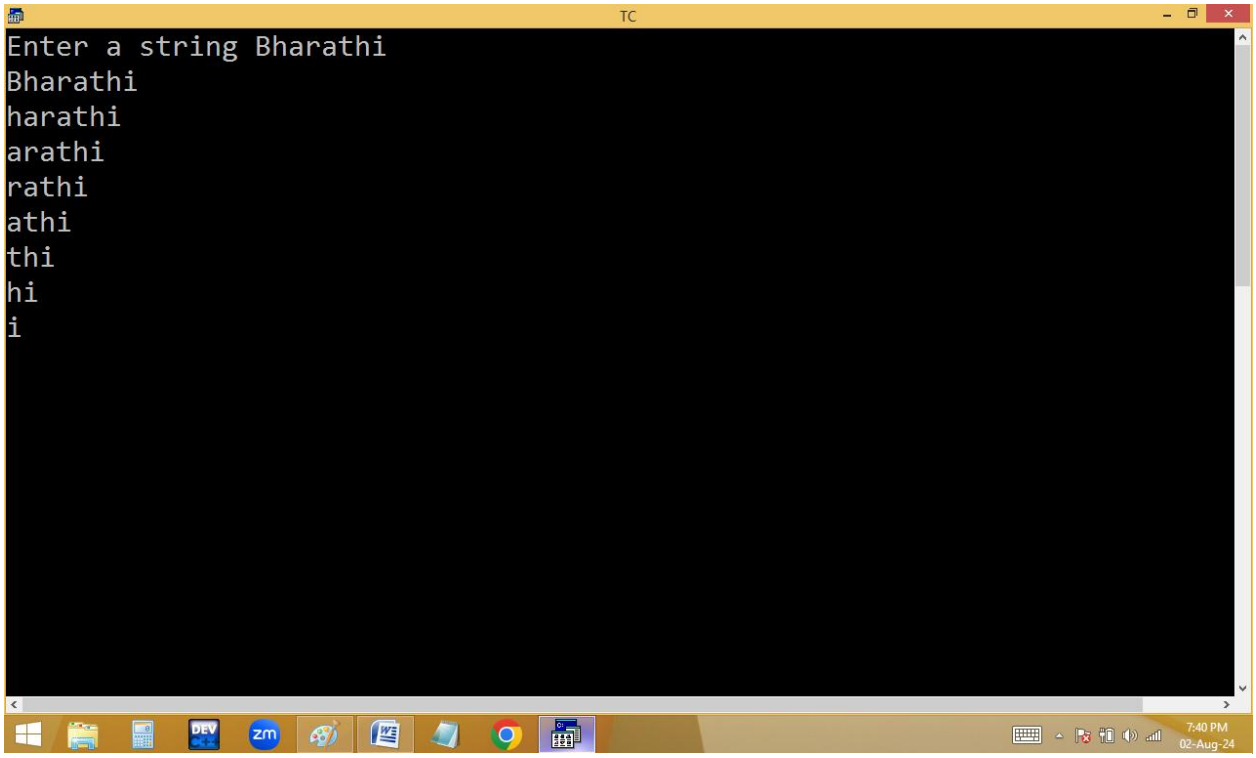


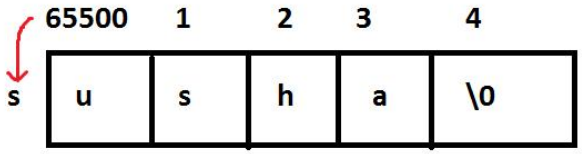


Reading and printing a string:


TC
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 41 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s[100]; int i;
clrscr();
printf("Enter a string "); scanf("%s",s);
for(i=0;s[i]!='\0';i++)printf("%s\n",s+i);
getch();
}


TC
Enter a string Priya
Priya
riya
iya
ya
a






```
for(i=0; s[i]!='\0';i++) printf("%s\n",s+i);
```

usha

usha
ush
us
u

65500+0*1=65500 to \0 ==> usha
65500+1*1=65501 to \0 ==> sha
65500+2*1=65502 to \0 ==> ha
65500+3*1=65503 to \0 ==> a
65500+4*1=65504 ==> \0 != \0 ==> false