

3-dimensional arrays:

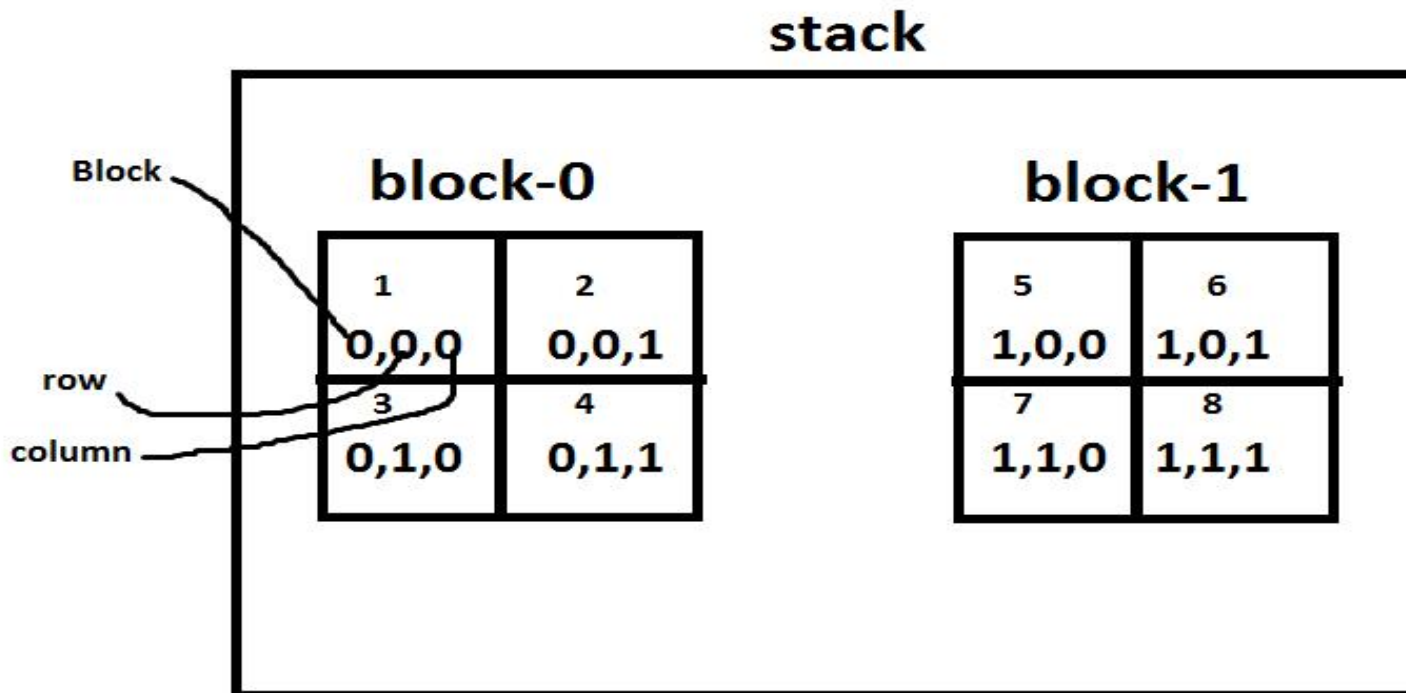
An array with several blocks, rows and columns.

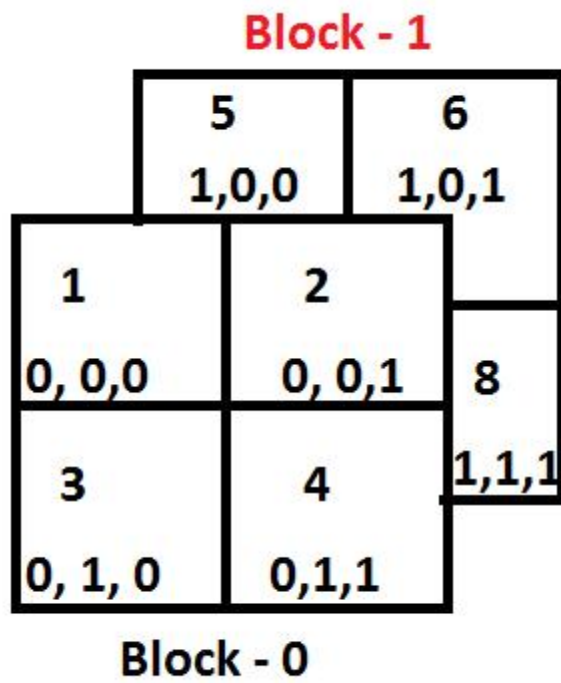
An array with 3 subscripting operators **[] [] []**.

Syntax:

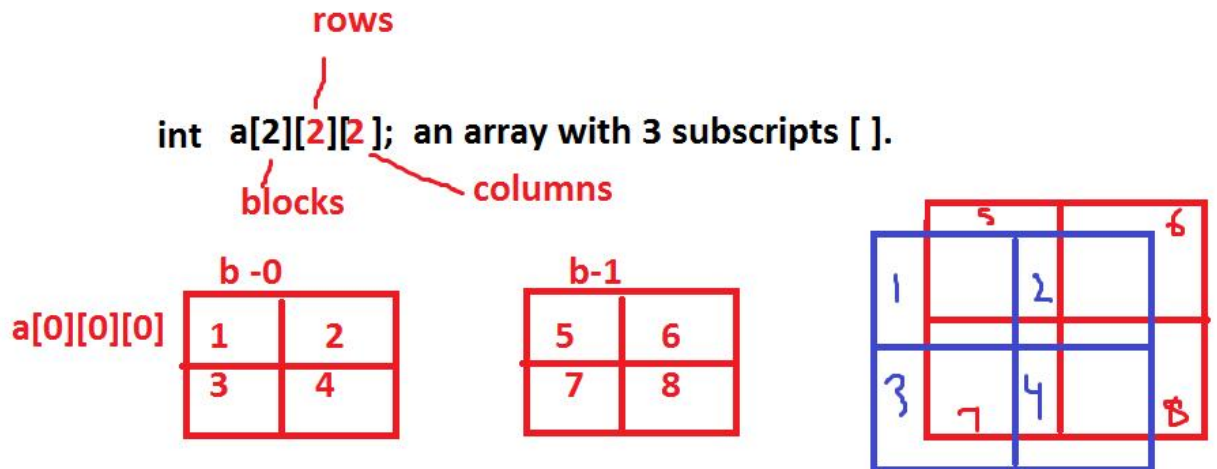
datatype variable [blocks] [rows] [columns] ;

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

```
TC
0-Block
  1  2
  3  4
1-Block
  5  6
  7  8
```

4-dimensional array:

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

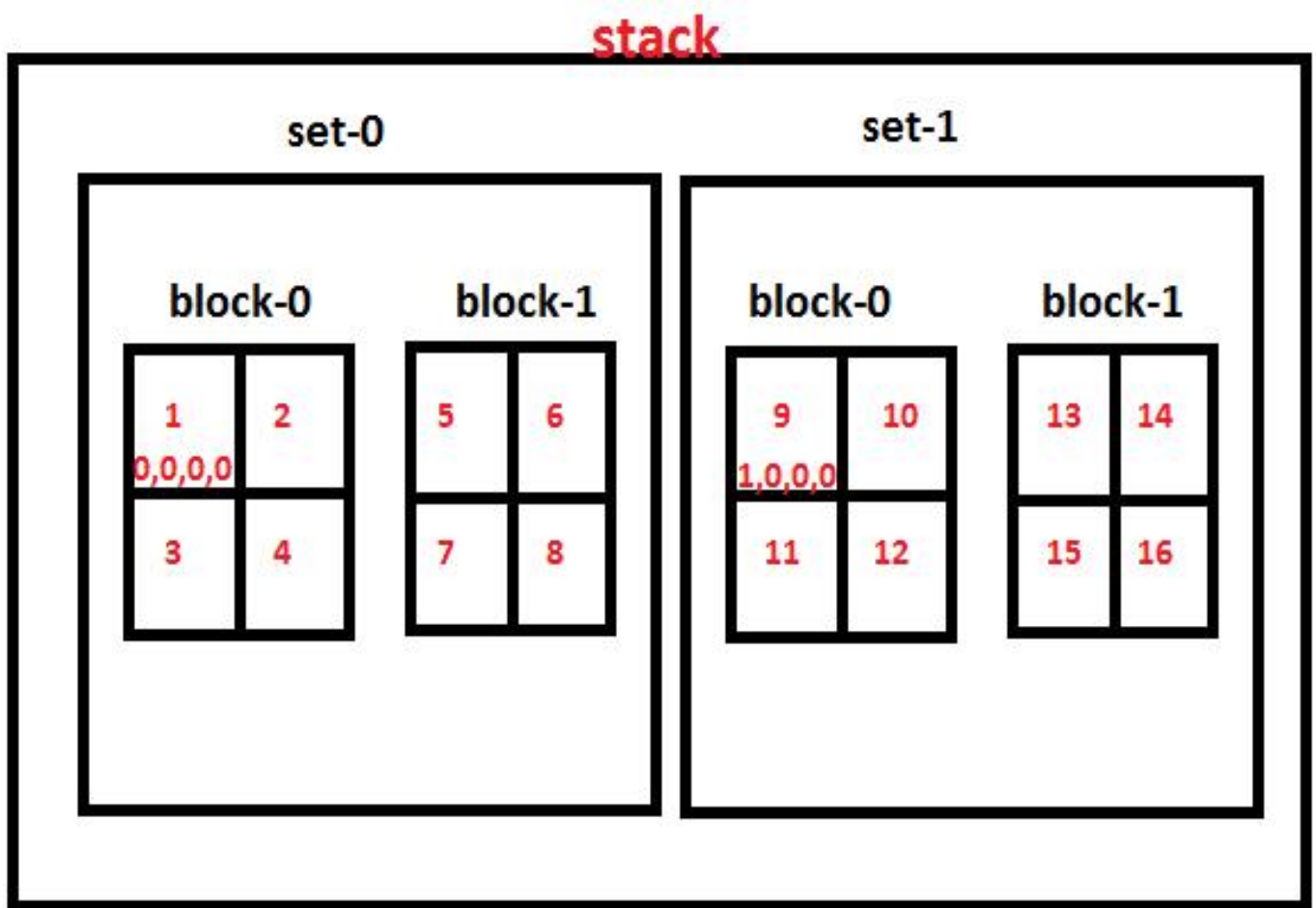
An array with 4 subscripting operators **[] [] [] []**.

Syntax:

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

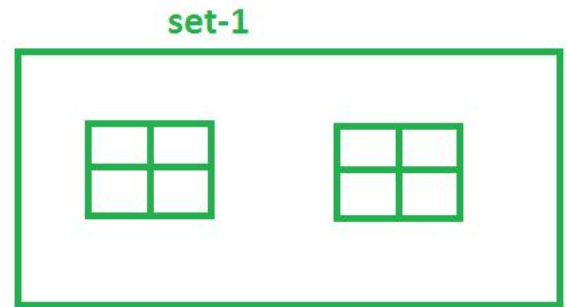
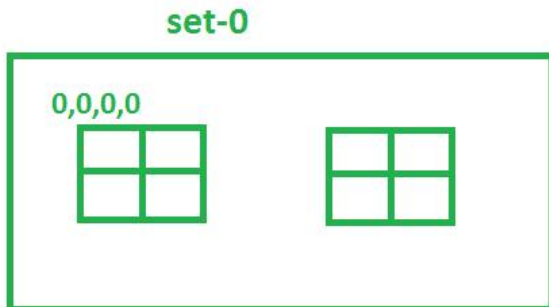
eg:

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];
```

set
int a[2][2][2][2]; an array with 3 subscripts [].
rows
blocks
columns



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.

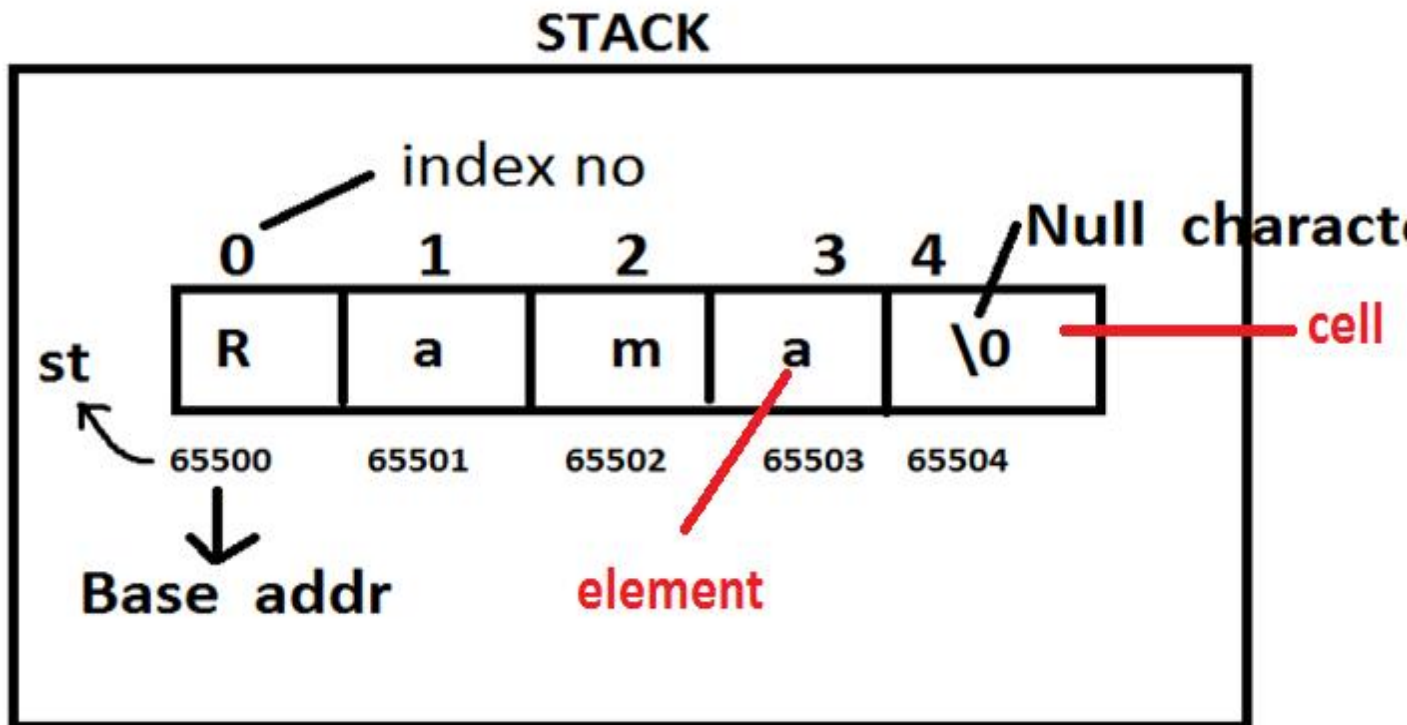
Syntax:

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

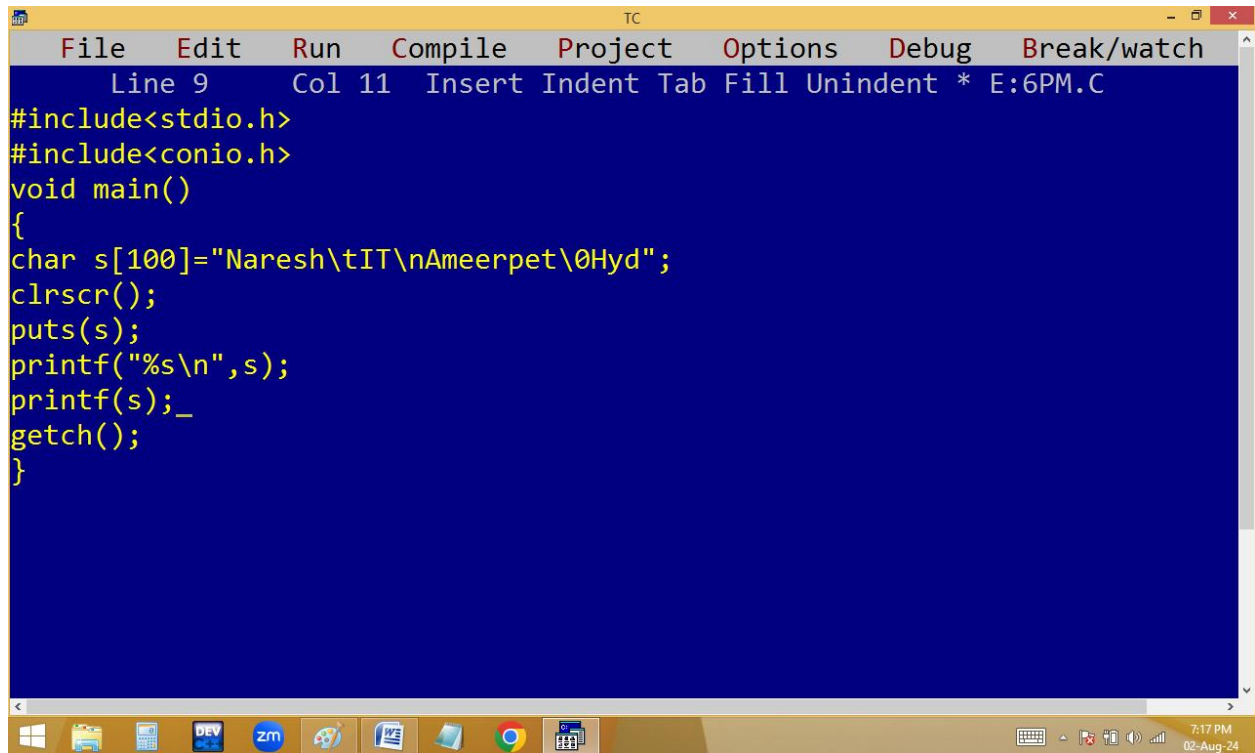
#define n 20

char st[n]; Ok

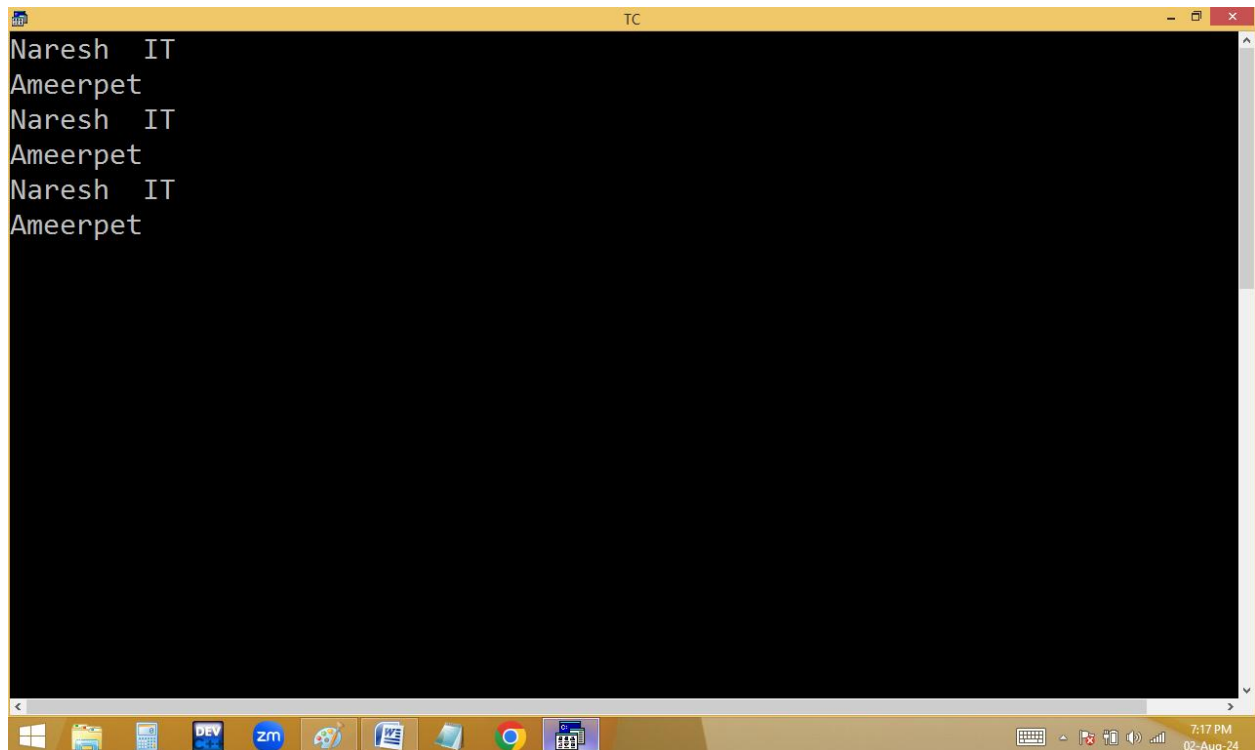
Note: String variable size always constant positive integer value.

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();
}
```

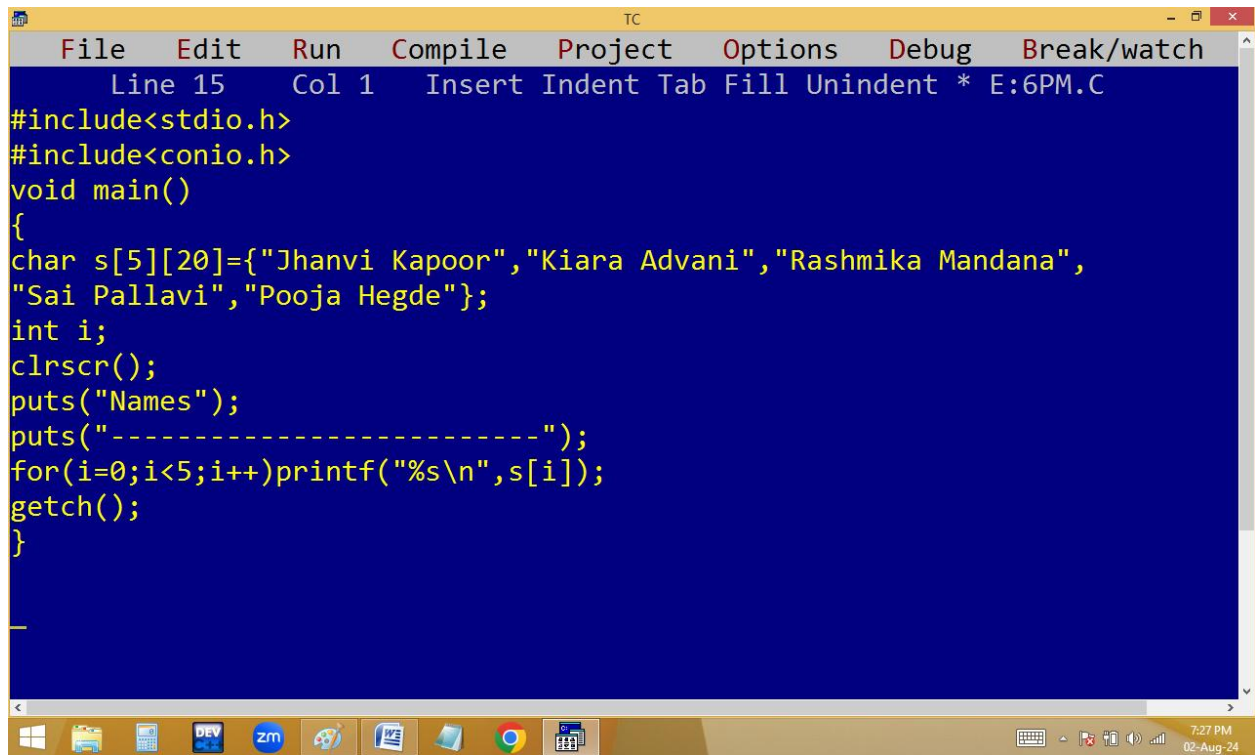


```
TC
Naresh IT
Ameerpet
Naresh IT
Ameerpet
Naresh IT
Ameerpet
```

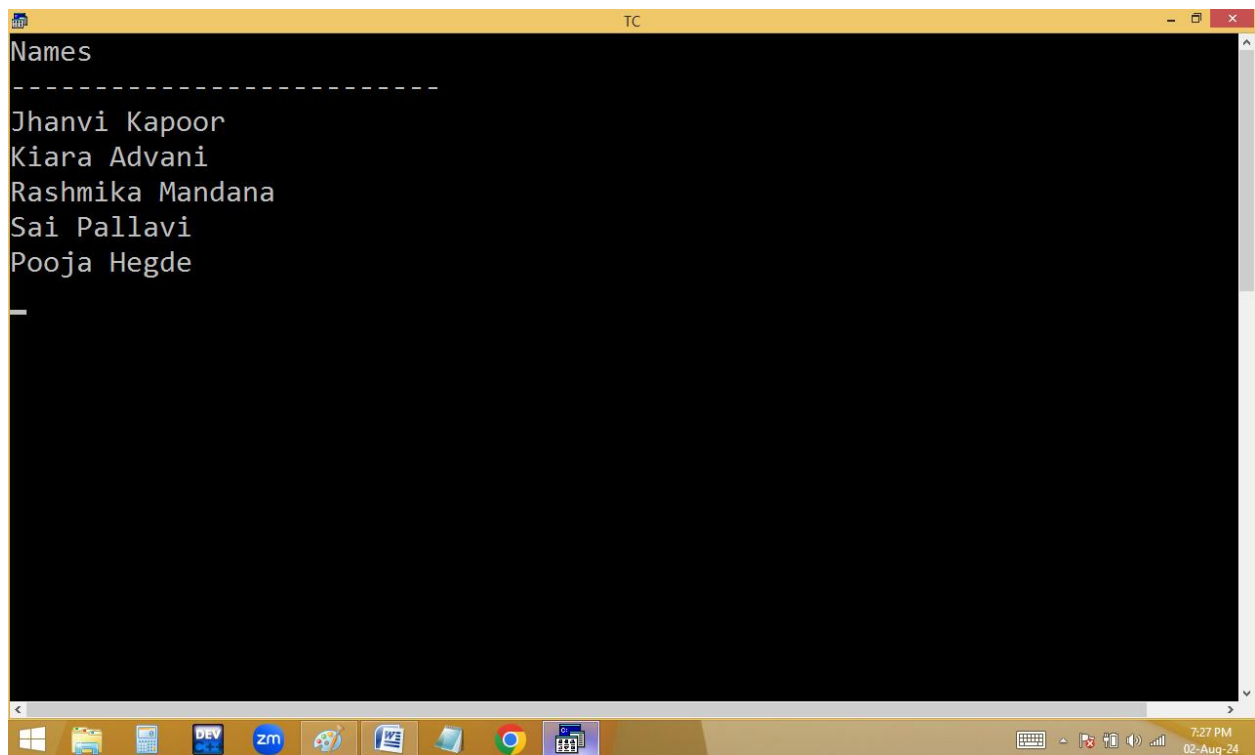
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 8 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s1[2]="N",s2[]="N",s3[2]={'N'},s4[]={'N','\0'},s5[]={'N'};
clrscr();
puts(s1);
puts(s2);
puts(s3);
puts(s4);
puts(s5);
getch();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

```
TC
N
N
N
N
NC∞ ↔⊕⊕
```

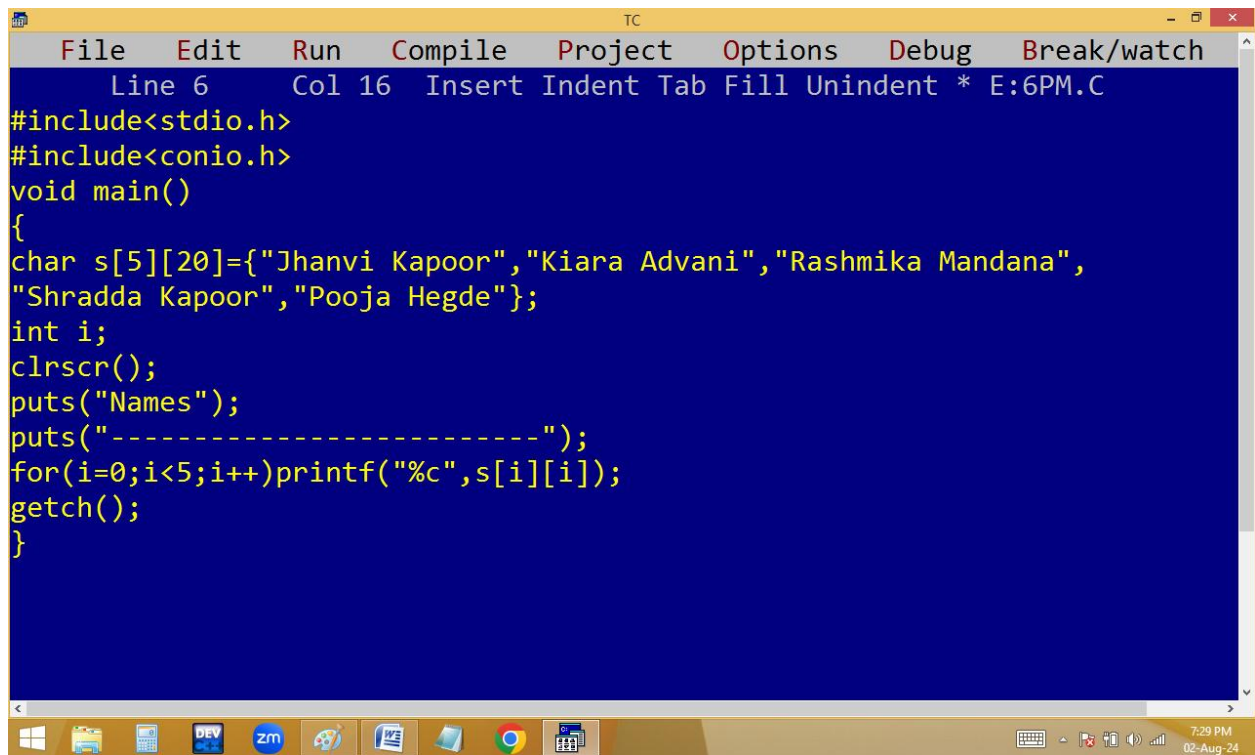
Storing of multiple strings:



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 15 Col 1 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",
"Sai Pallavi","Pooja Hegde"};
int i;
clrscr();
puts("Names");
puts("-----");
for(i=0;i<5;i++)printf("%s\n",s[i]);
getch();
}
```



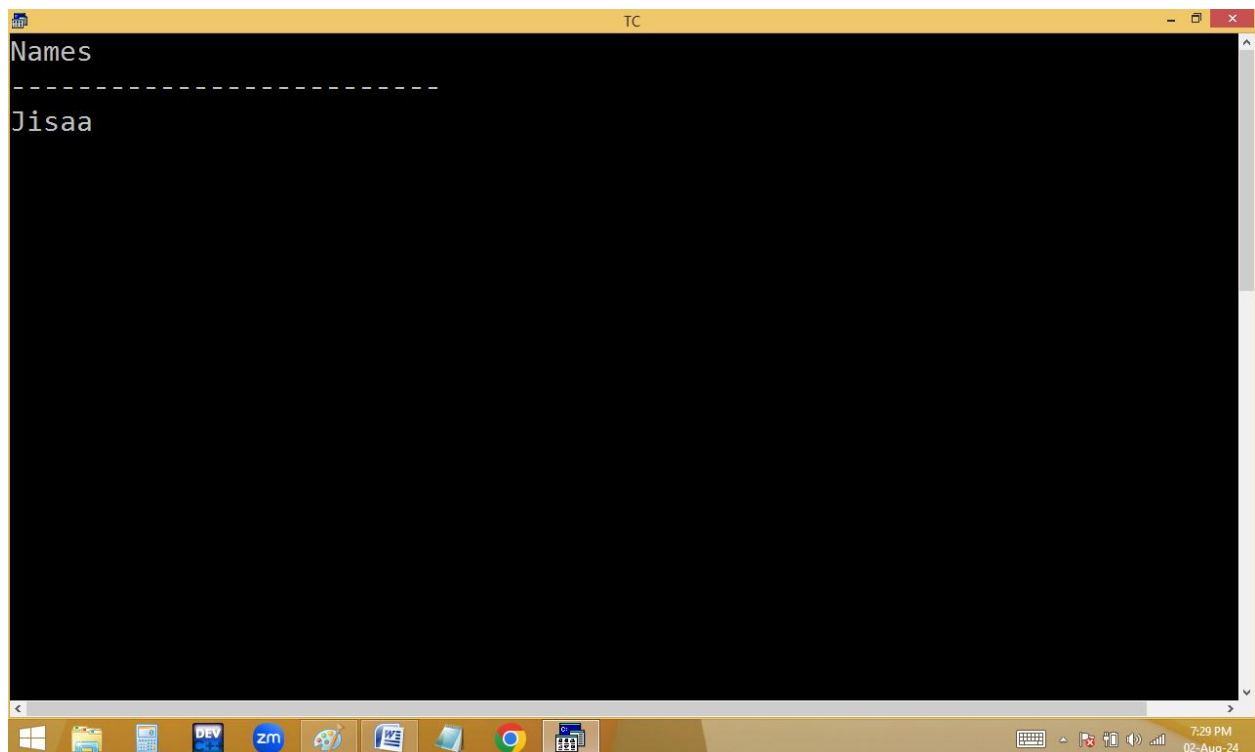
```
TC
Names
-----
Jhanvi Kapoor
Kiara Advani
Rashmika Mandana
Sai Pallavi
Pooja Hegde
```



The screenshot shows the Turbo C++ (TC) IDE with a blue background. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 6', 'Col 16', and 'Insert' mode. The code in the editor is as follows:

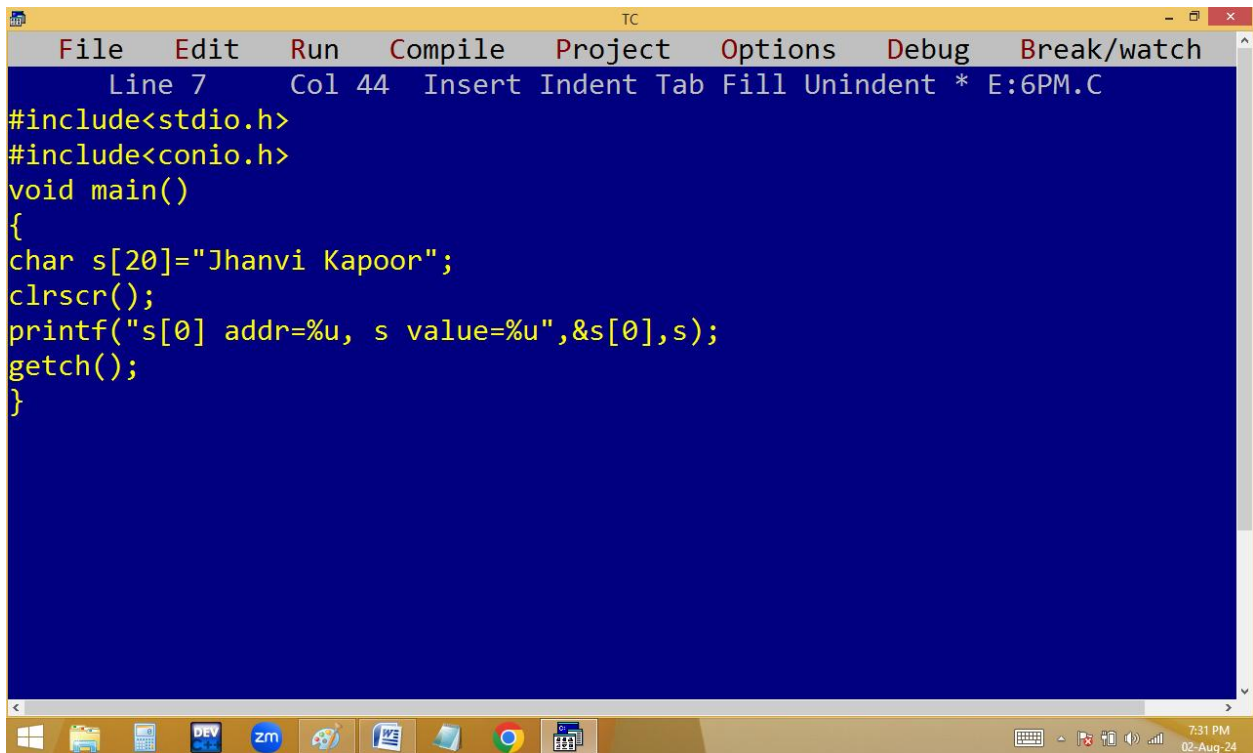
```
Line 6      Col 16      Insert      Indent      Tab      Fill      Unindent      *      E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s[5][20]={"Jhanvi Kapoor","Kiara Advani","Rashmika Mandana",
"Shradda Kapoor","Pooja Hegde"};
int i;
clrscr();
puts("Names");
puts("-----");
for(i=0;i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

The Windows taskbar at the bottom shows various icons including the Start button, File Explorer, Calculator, DEV C++, Zoom, and others. The system clock in the bottom right corner displays '7:29 PM' and '02-Aug-24'.

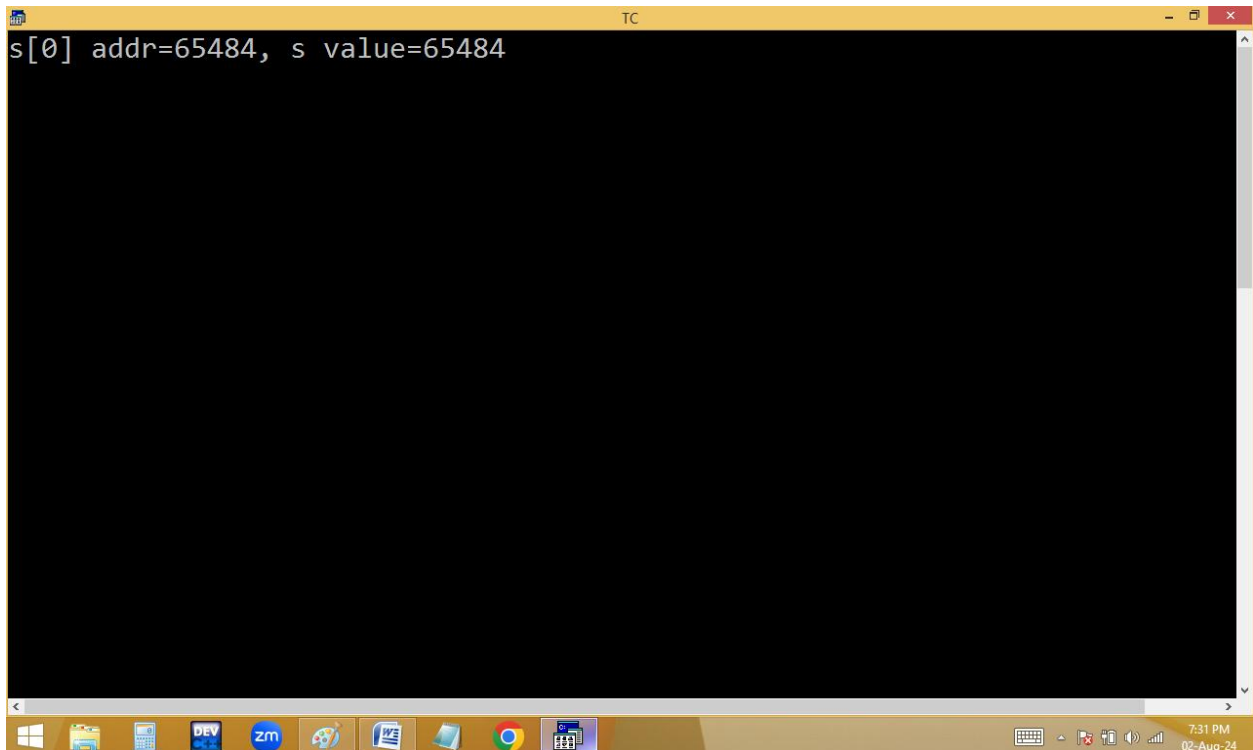


The screenshot shows the Turbo C++ (TC) IDE with a black background, displaying the output of the program. The text 'Names' is printed on the first line, followed by a dashed line '-----' on the second line. The output is partially cut off, showing 'Jisaa' on the third line. The menu bar and status bar are visible at the top, and the Windows taskbar is at the bottom, showing the same system clock as the previous screenshot.

Finding string address:

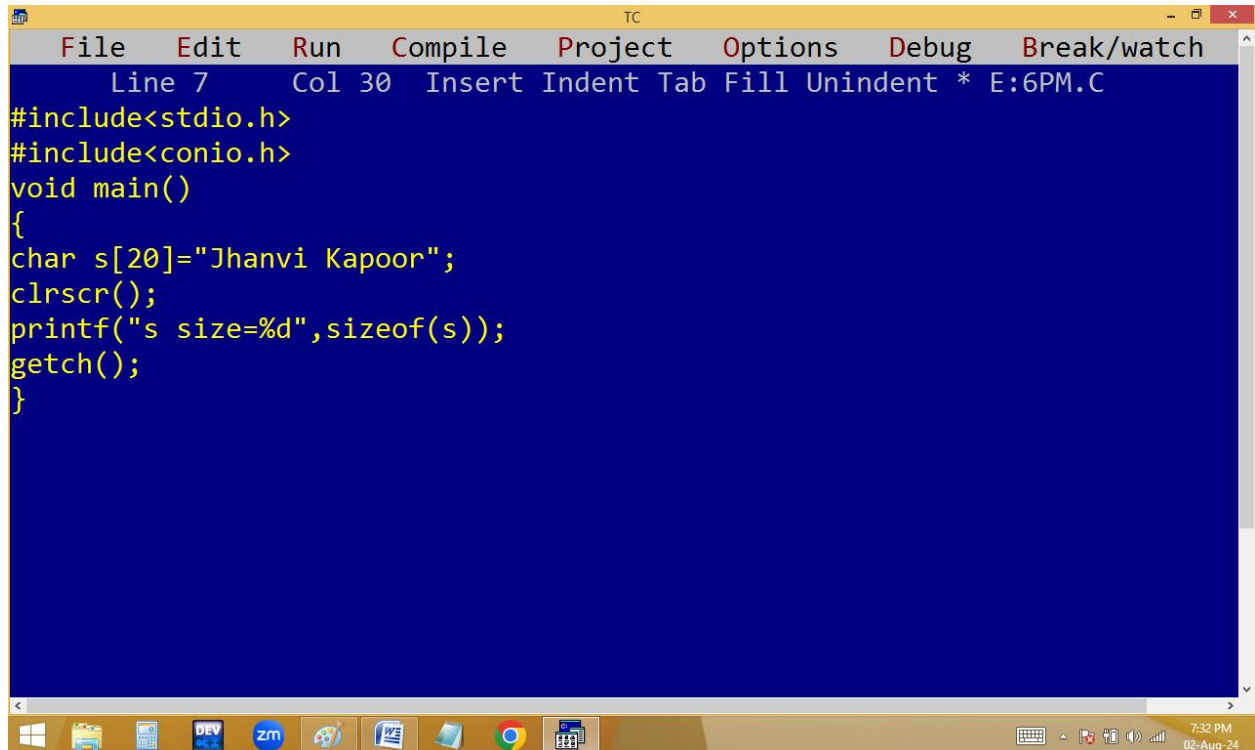


```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 7 Col 44 Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char s[20]="Jhanvi Kapoor";
clrscr();
printf("s[0] addr=%u, s value=%u",&s[0],s);
getch();
}
```



```
TC
s[0] addr=65484, s value=65484
```

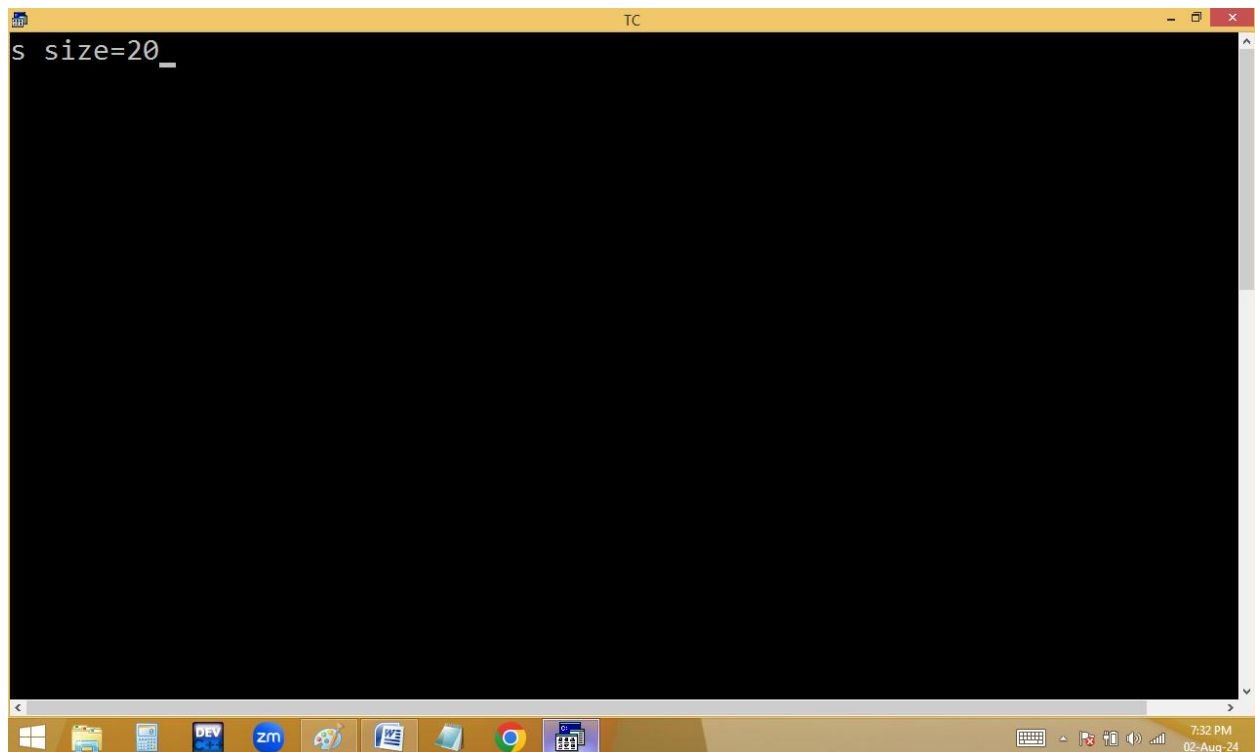
Finding string size:



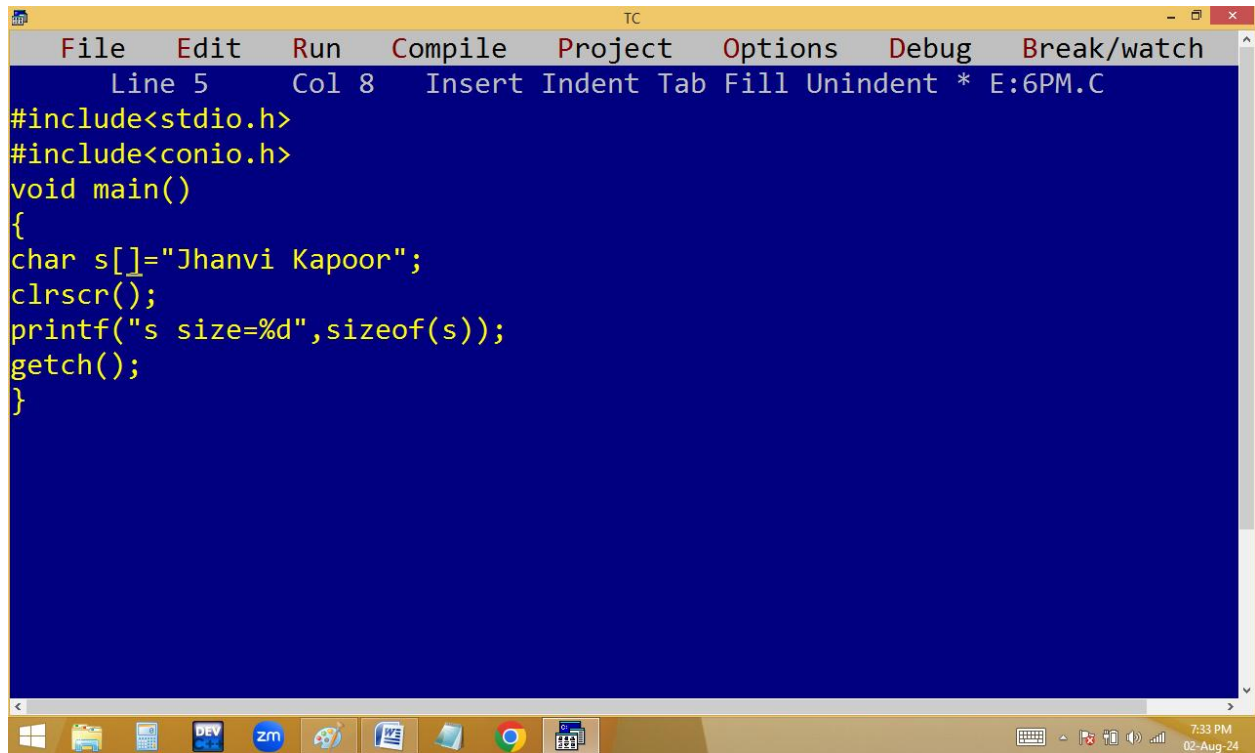
The screenshot shows the Turbo C++ (TC) IDE with a blue background. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 7 Col 30 Insert Indent Tab Fill Unindent * E:6PM.C'. The code in the editor is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
char s[20]="Jhanvi Kapoor";
clrscr();
printf("s size=%d",sizeof(s));
getch();
}
```

The Windows taskbar at the bottom shows various application icons and the system clock indicating 7:32 PM on 02-Aug-24.



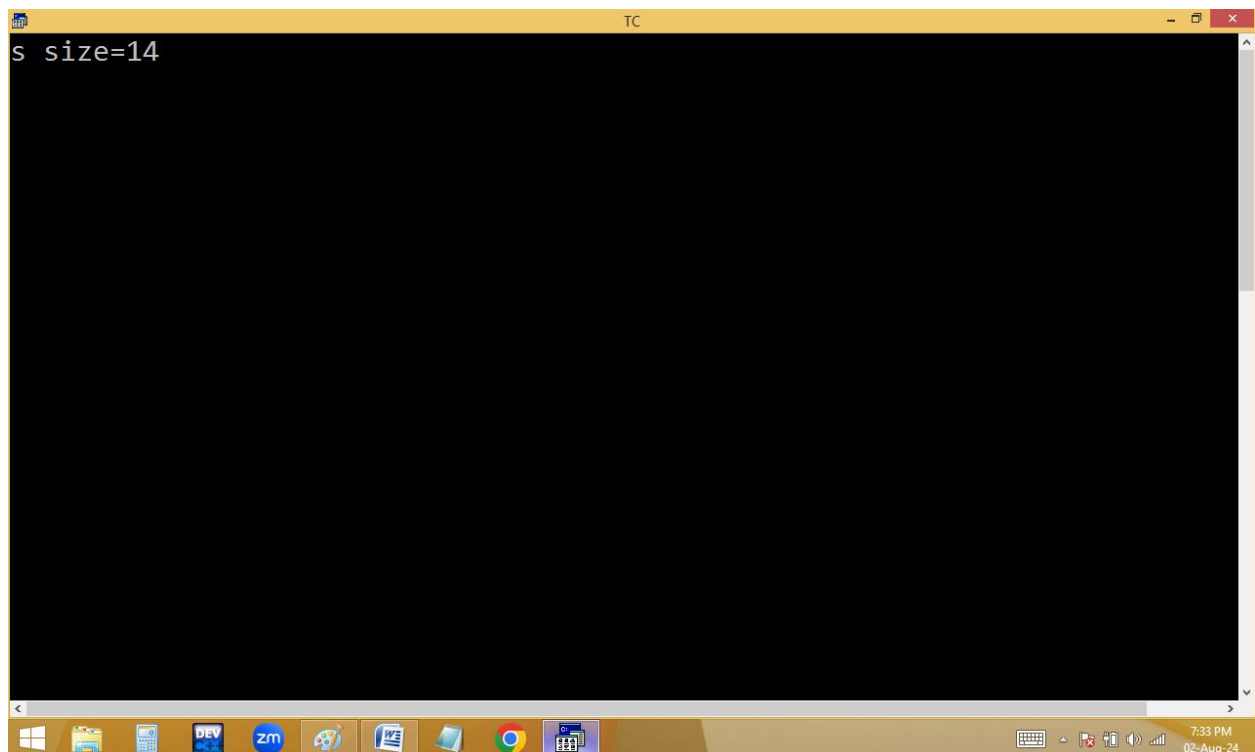
The screenshot shows the Turbo C++ (TC) IDE with a black background, displaying the output of the program. The text 's size=20_' is visible on the first line. The Windows taskbar at the bottom is the same as in the previous screenshot, showing the system clock at 7:32 PM on 02-Aug-24.



The screenshot shows the Turbo C++ (TC) IDE with a blue background. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 5 Col 8' and 'Insert Indent Tab Fill Unindent * E:6PM.C'. The code in the editor is as follows:

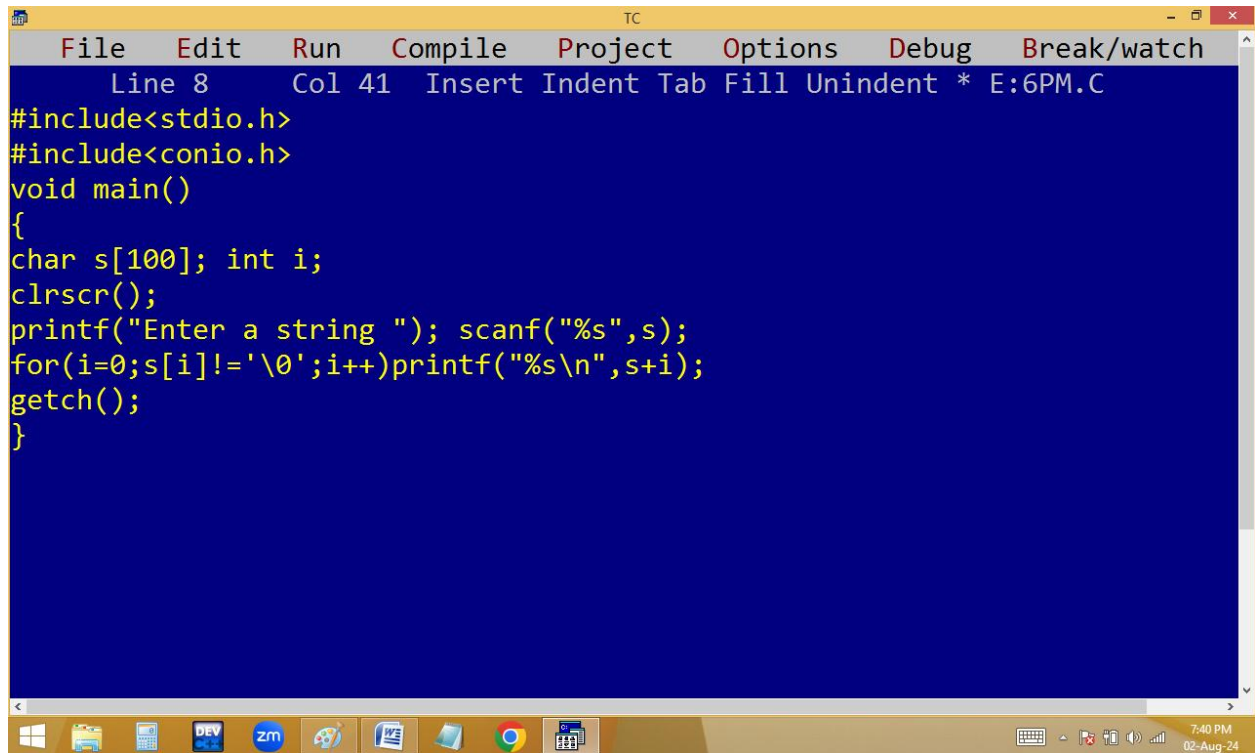
```
#include<stdio.h>
#include<conio.h>
void main()
{
char s[]="Jhanvi Kapoor";
clrscr();
printf("s size=%d",sizeof(s));
getch();
}
```

The Windows taskbar at the bottom shows various icons including Windows, File Explorer, DEV, zm, and others, along with the system clock showing 7:33 PM on 02-Aug-24.



The screenshot shows the Turbo C++ (TC) IDE with a black background. The output window displays the text 's size=14'. The Windows taskbar at the bottom is identical to the previous screenshot, showing the same icons and system clock.

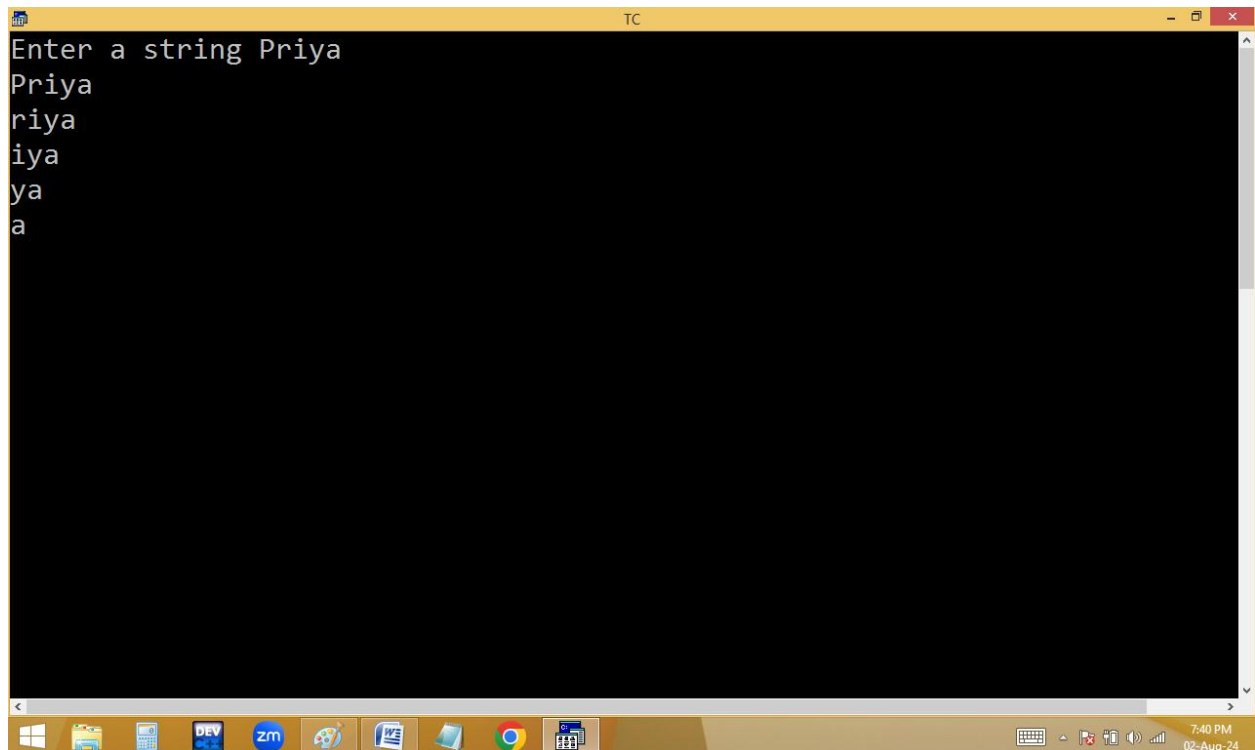
Reading and printing a string:



The screenshot shows the Turbo C++ (TC) IDE with a blue background. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 8 Col 41 Insert Indent Tab Fill Unindent * E:6PM.C'. The code in the editor is as follows:

```
#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();
}
```

The Windows taskbar at the bottom shows icons for Windows, File Explorer, Calculator, DEV C++, Zoom, Paint, Notepad, Chrome, and Task Manager. The system clock shows 7:40 PM on 02-Aug-24.

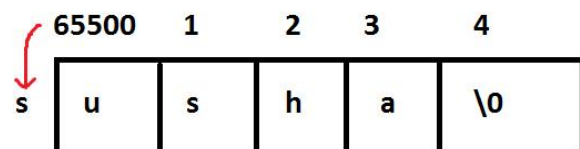


The screenshot shows the Turbo C++ (TC) IDE with a black background, displaying the output of the program. The text 'Enter a string Priya' is on the first line. Below it, the string 'Priya' is printed character by character on four separate lines: 'Priya', 'riya', 'iya', 'ya', and 'a'.

```
Enter a string Priya
Priya
riya
iya
ya
a
```

The Windows taskbar at the bottom is identical to the first screenshot, showing the same icons and system clock (7:40 PM on 02-Aug-24).

```
TC
Enter a string Bharathi
Bharathi
harathi
arathi
rathi
athi
thi
hi
i
```



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

↓

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