

Two dimensional arrays:

Array with several rows and columns.

Array with two subscripting operators **[][]**.

It is array of arrays. i.e. collection of one-dimensional arrays.

It is implicit double pointer.

It is a $n \times n$ matrix.

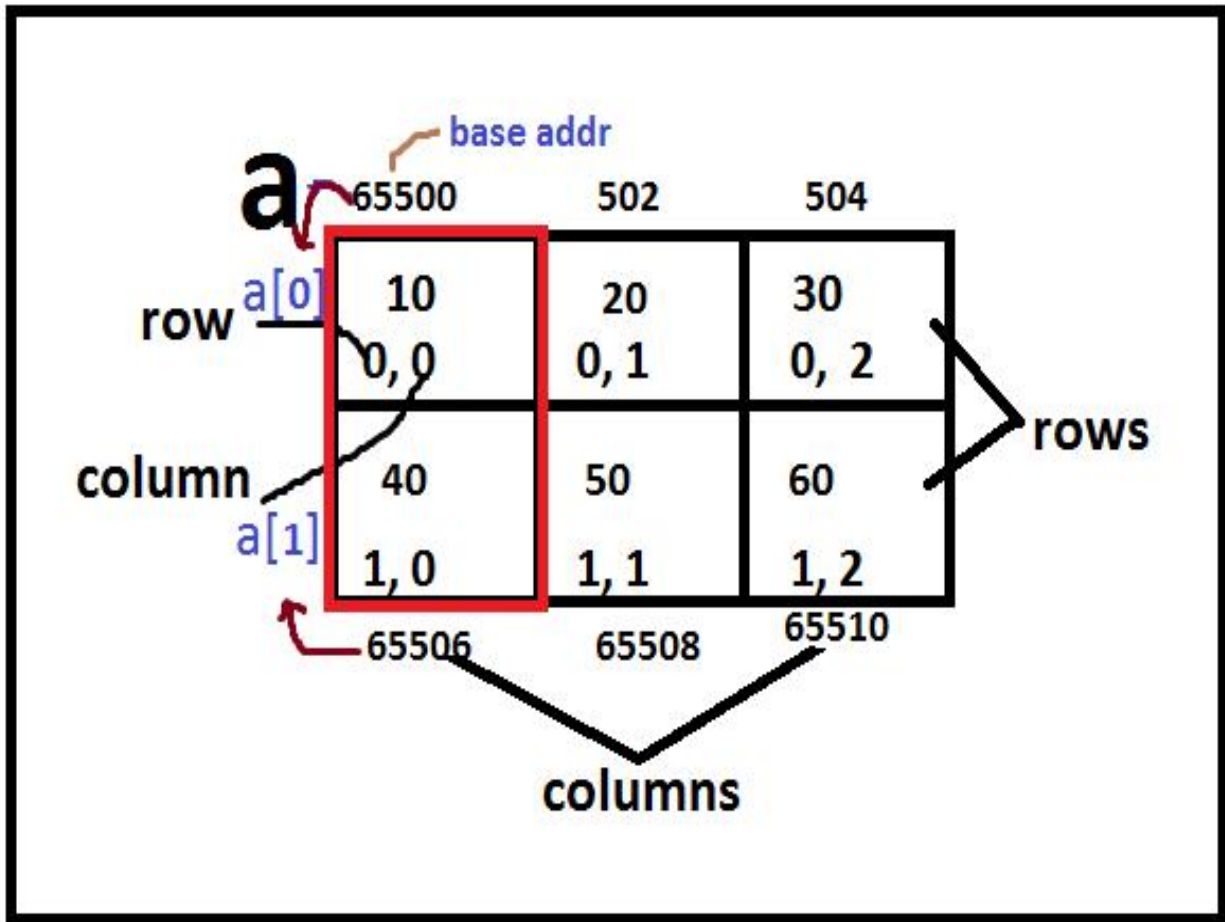
Syntax:

datatype variable [rows] [columns] = {elements} ;

Eg:

```
int a[ 2 ][ 3 ] = { { 10, 20 , 30 } , { 40 ,50, 60 } };
```

stack



In two dimensional array the rows/first subscript is working as array of pointers and they stores first column address of each row. Hence it is an implicit/internal double pointer.

In the above example, To print the first row, first column value, we have to use

`printf("%d", a[0][0]);` → 10

Internally how this statement is working ?

`a[0]` means value at `a[0]` i.e. 65500.

$65500 + [0] \text{ col} \rightarrow 65500 + 0 * 2 \rightarrow 65500 \rightarrow$
value at 65500 is 10.

Index no

Int size

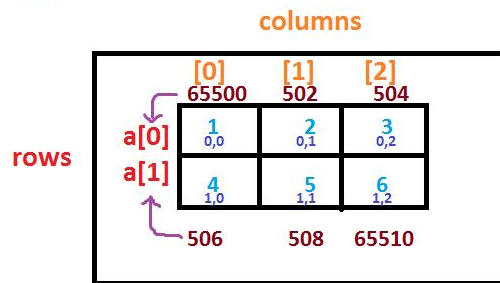
stack

<code>a[1][2]</code>	<u>60</u>	65500
<code>a[1][1]</code>	<u>50</u>	65498
<code>a[1][0]</code>	40	65496
<code>a[0][2]</code>	<u>30</u>	65494
<code>a[0][1]</code>	<u>20</u>	65492
<code>a[0][0]</code>	<u>10</u>	65490

`int a[2][2]={10,20,30,40};`

<code>a[1][1]</code>	40	65496
<code>a[1][0]</code>	<u>30</u>	65494
<code>a[0][1]</code>	<u>20</u>	65492
<code>a[0][0]</code>	<u>10</u>	65490

int a[2][3]={1,2,3},{4,5,6};



p(a[0][0]); ==> 1

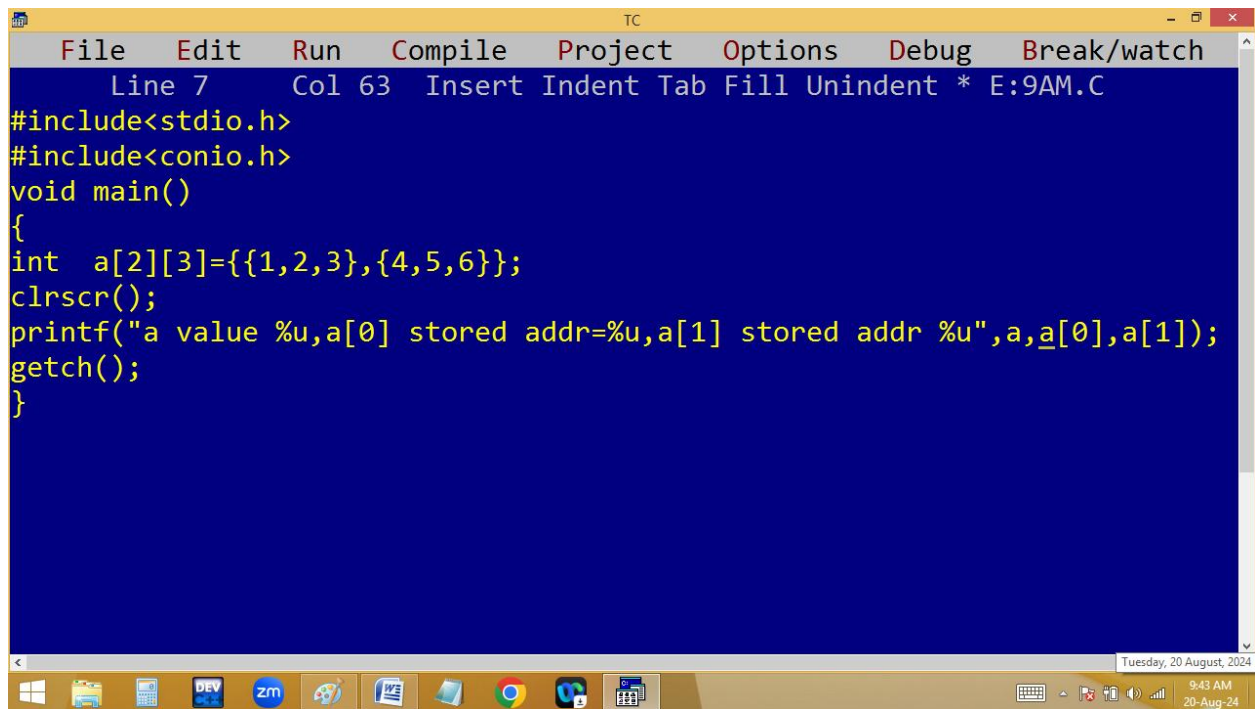
65500+0*2=65500 ==> print the value at 65500 ==> 1

p(a[1][2]) ==> 6

65506+2*2=65510 ==> print the value at 65510 ==> 6

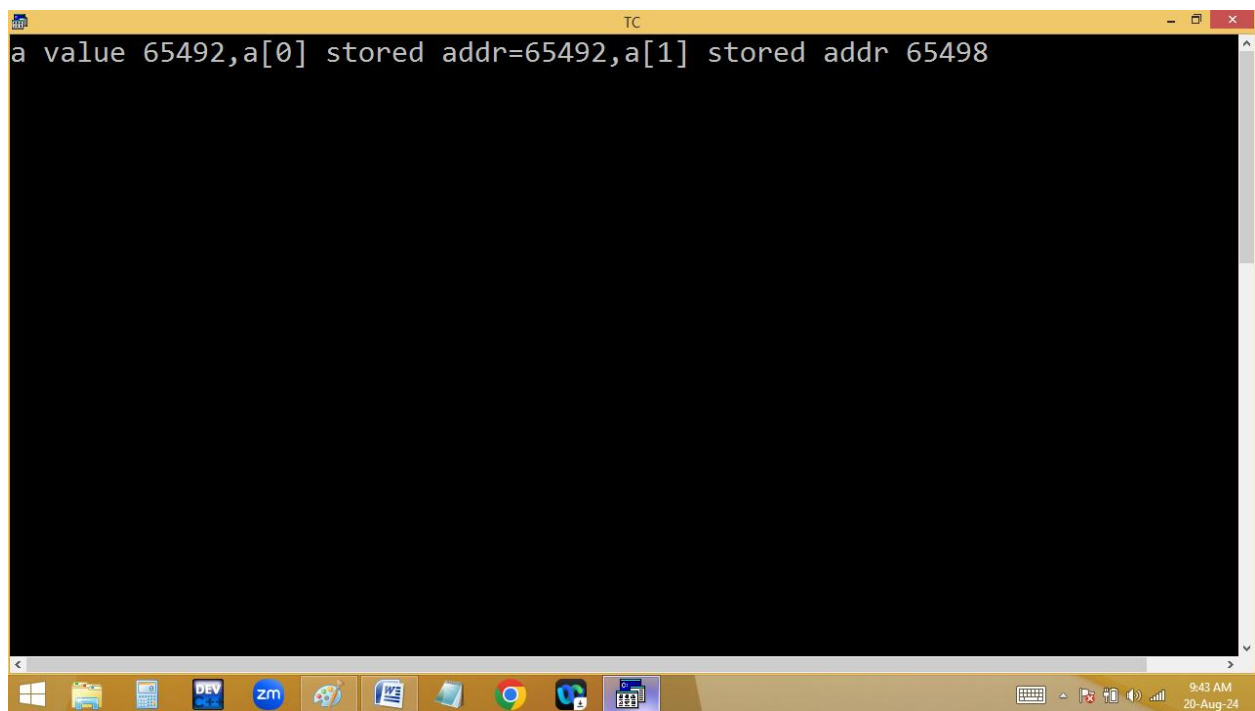
a[1][2]	6	65510
a[1][1]	5	65508
a[1][0]	4	65506
a[0][2]	3	65504
a[0][1]	2	65502
a[0][0]	1	65500

Finding array address:



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 7 Col 63 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[2][3]={{1,2,3},{4,5,6}};
clrscr();
printf("a value %u,a[0] stored addr=%u,a[1] stored addr %u",a,a[0],a[1]);
getch();
}
```

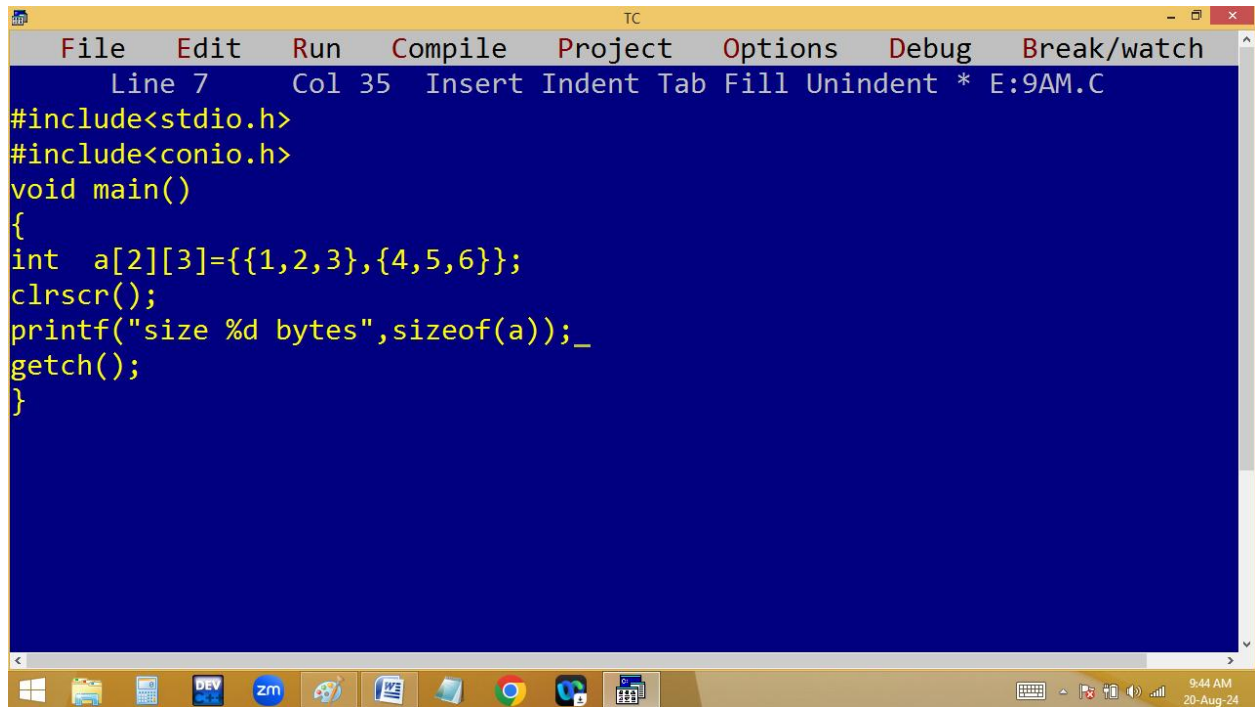
Tuesday, 20 August, 2024 9:43 AM 20-Aug-24



```
TC
a value 65492,a[0] stored addr=65492,a[1] stored addr 65498
```

9:43 AM 20-Aug-24

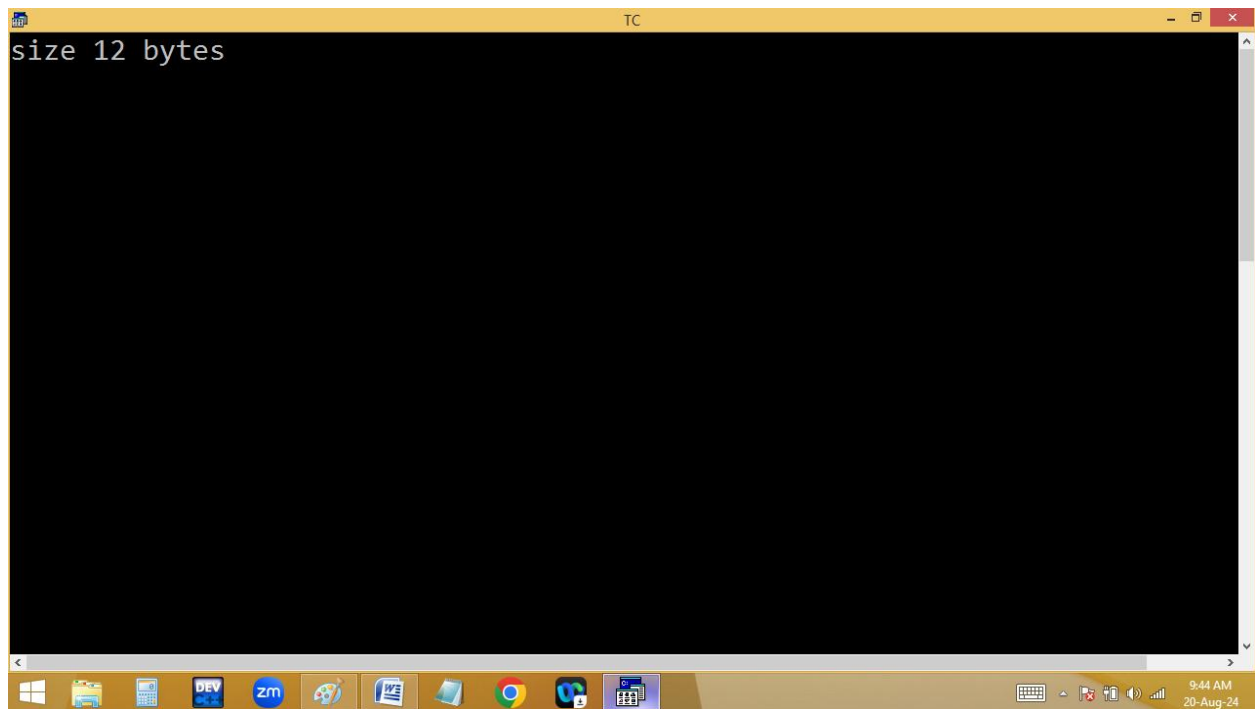
Finding $n*n$ matrix size:



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 35 Insert Indent Tab Fill Unindent * E:9AM.C'. The code in the editor is as follows:

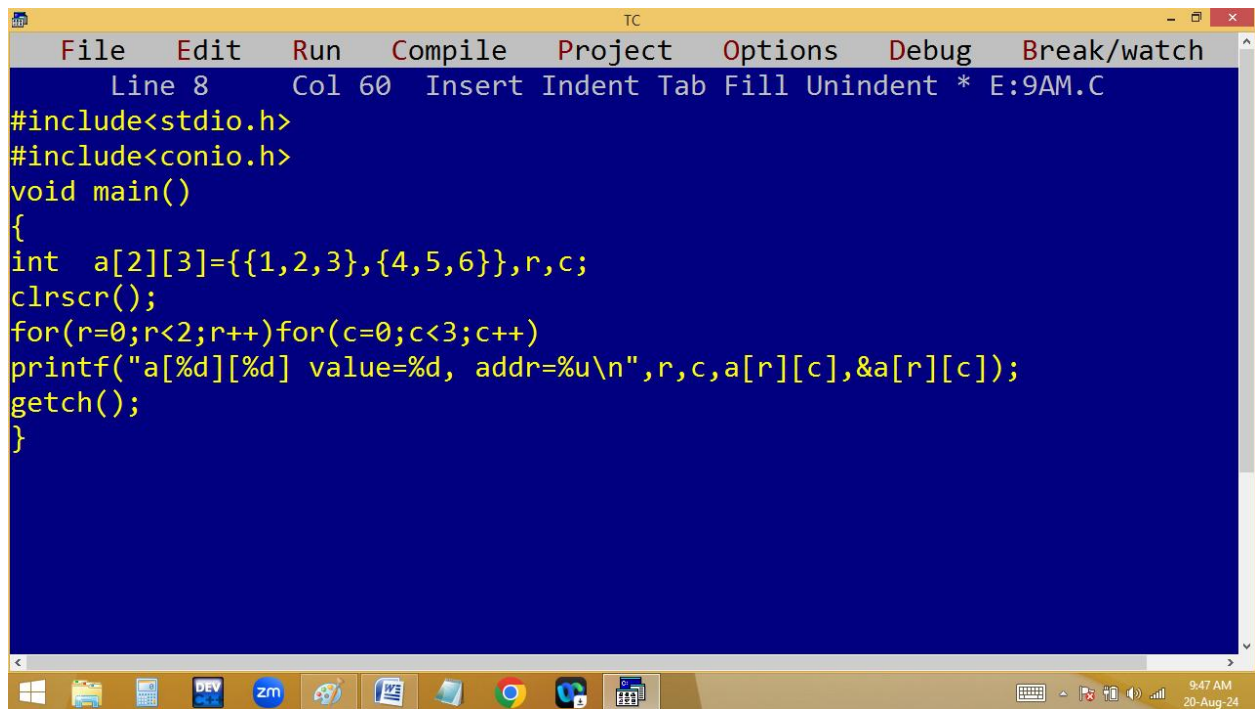
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={1,2,3},{4,5,6}};
clrscr();
printf("size %d bytes",sizeof(a));_
getch();
}
```

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 shows 9:44 AM on 20-Aug-24.

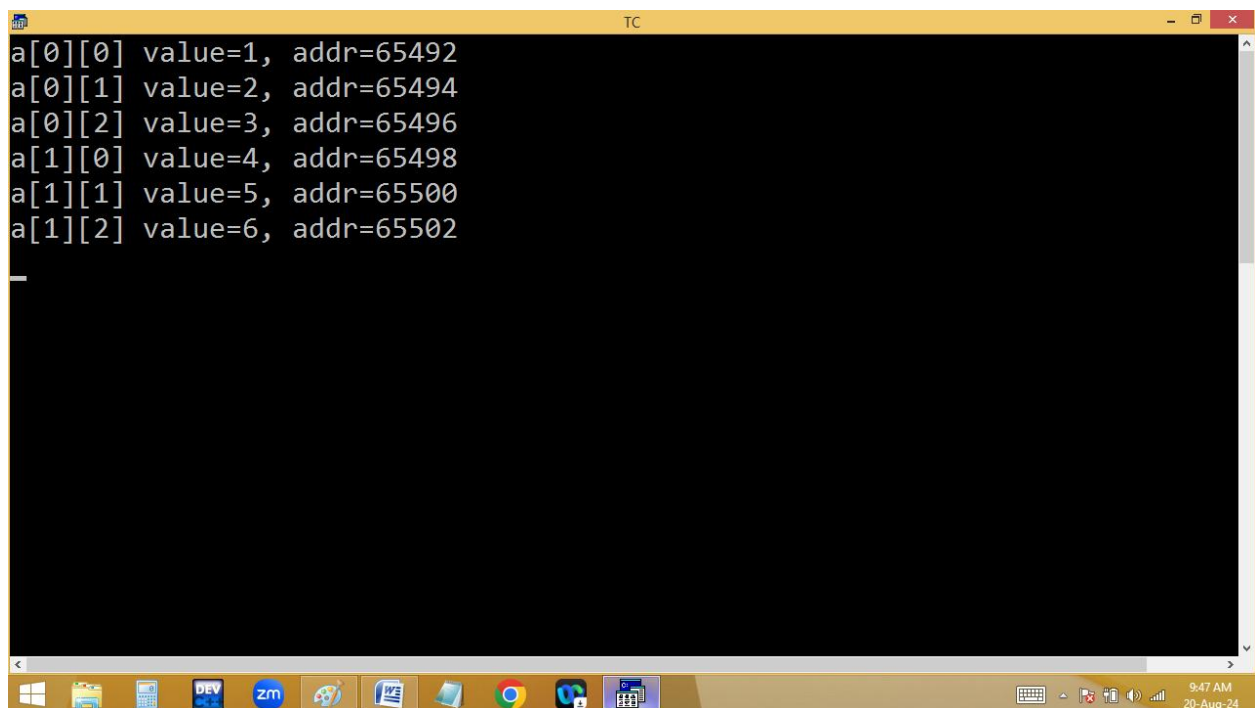


The screenshot shows the Turbo C++ (TC) IDE with a black background. The output window displays the text 'size 12 bytes'. The Windows taskbar at the bottom is identical to the first screenshot, showing the same icons and system clock (9:44 AM on 20-Aug-24).

Finding array value, address and position:



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 60 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[2][3]={{1,2,3},{4,5,6}},r,c;
clrscr();
for(r=0;r<2;r++)for(c=0;c<3;c++)
printf("a[%d][%d] value=%d, addr=%u\n",r,c,a[r][c],&a[r][c]);
getch();
}
```



```
TC
a[0][0] value=1, addr=65492
a[0][1] value=2, addr=65494
a[0][2] value=3, addr=65496
a[1][0] value=4, addr=65498
a[1][1] value=5, addr=65500
a[1][2] value=6, addr=65502
```

Direct initialization of a n*n matrix:

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

```
TC
Elements
 1  2  3
 4  5  6
```



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

Windows taskbar: 9:50 AM, 20-Aug-24

```
TC
Elements
 1  2  3
 4  5  6
```

Windows taskbar: 9:51 AM, 20-Aug-24

```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={1,2},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```

```
TC
Elements
 1  2  0
0  0  0
```

```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},{2}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```

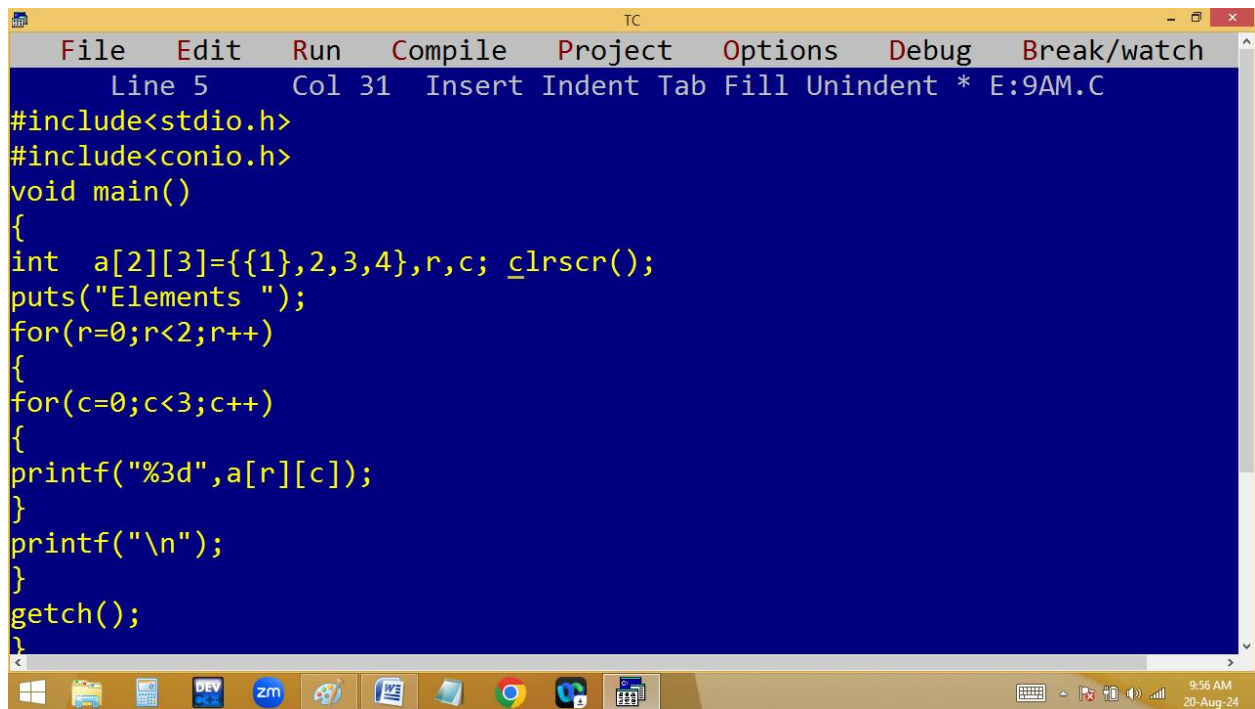
9:53 AM
20-Aug-24

```
TC
Elements
1  0  0
2  0  0
```

9:54 AM
20-Aug-24

```
TC
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},{2},{3}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```

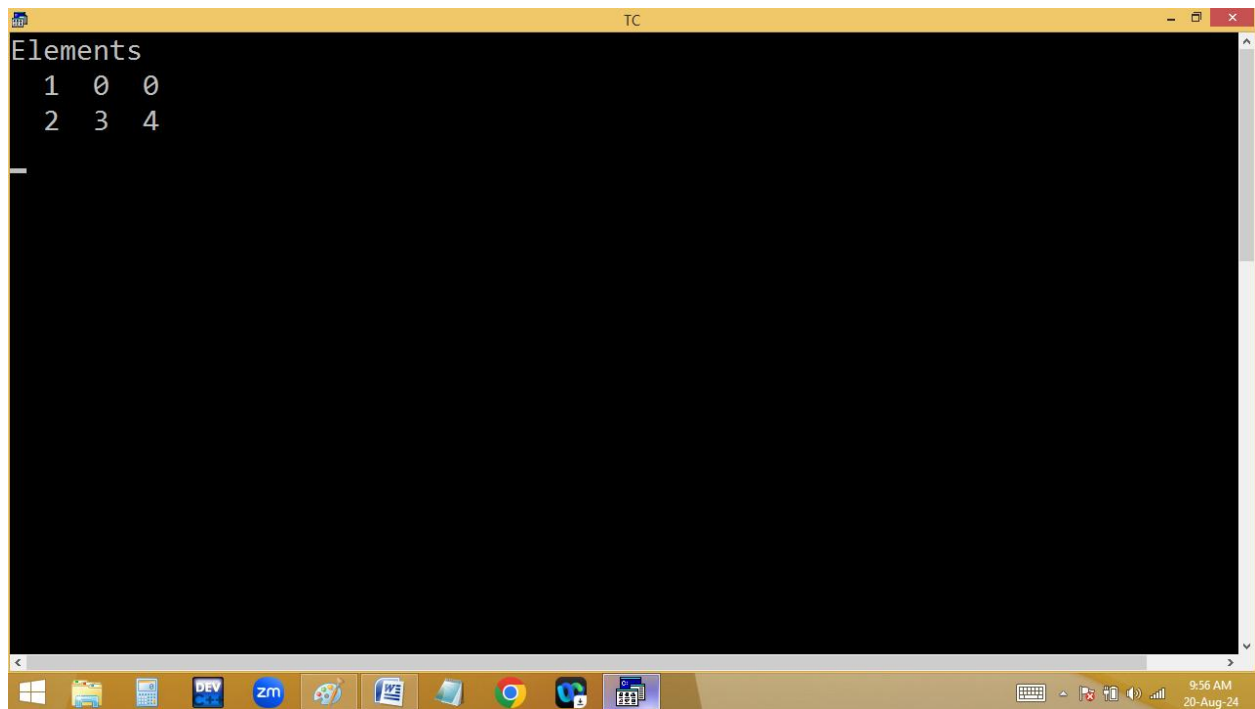
```
TC
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2,3,4}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



The screenshot shows the Turbo C++ (TC) IDE with a yellow title bar and menu bar. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 5', 'Col 31', and 'Insert Indent Tab Fill Unindent * E:9AM.C'. The main editing area has a blue background and contains the following C code:

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

The Windows taskbar at the bottom shows various icons including the Start button, File Explorer, Calculator, and several application icons. The system clock in the bottom right corner displays '9:56 AM' and '20-Aug-24'.



The screenshot shows the Turbo C++ (TC) IDE with a yellow title bar. The main editing area has a black background and displays the output of the program:

```
Elements
1  0  0
2  3  4
```

The Windows taskbar at the bottom is identical to the first screenshot, showing the same application icons and system clock.

TC

Error: Too many initializers in function main

```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={1},{2},3,4},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```

9:57 AM
20-Aug-24

TC

File Edit Run Compile Project Options Debug Break/watch

Line 5 Col 26 Insert Indent Tab Fill Unindent * E:9AM.C

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

9:58 AM
20-Aug-24

```
TC
Elements
1 0 0
2 3 4
```

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 26 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[2][3]={{1,2},{3,4}},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



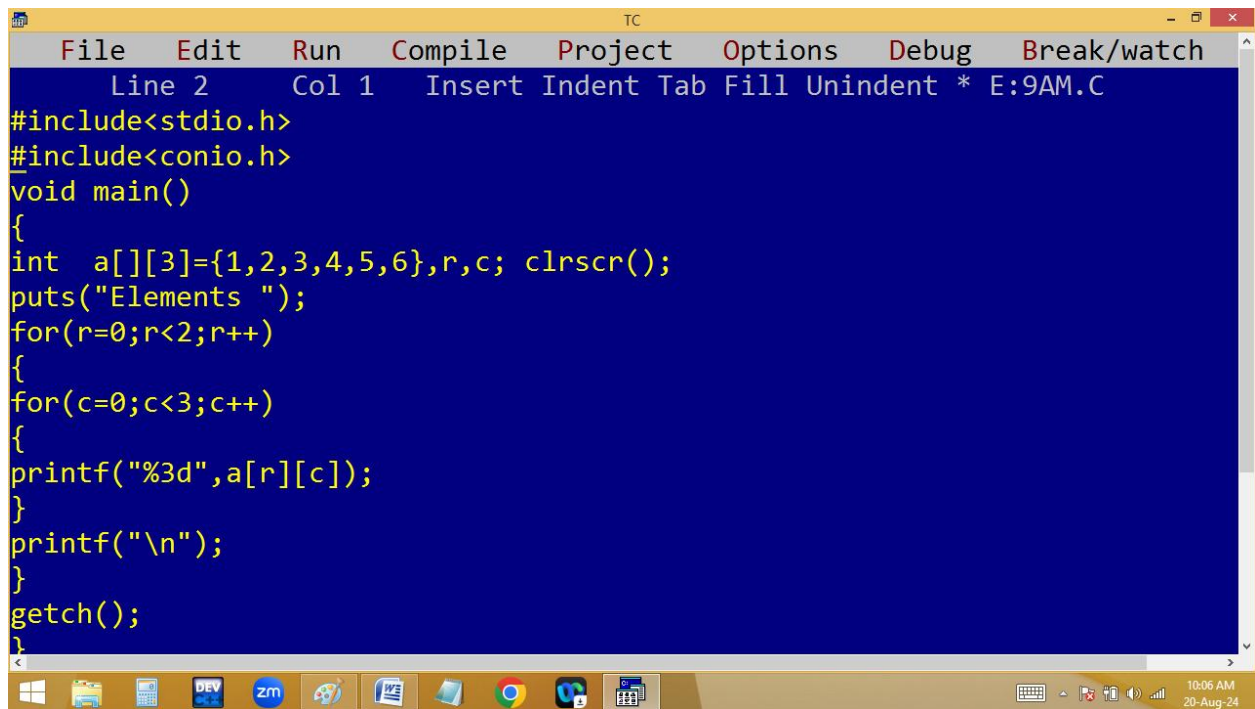
```
Elements
1  2  0
3  4  0
```

```
Error: Initializer syntax error in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={ {1,2},{ } },r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



```
TC
Error: Size of structure or array not known in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[][]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```

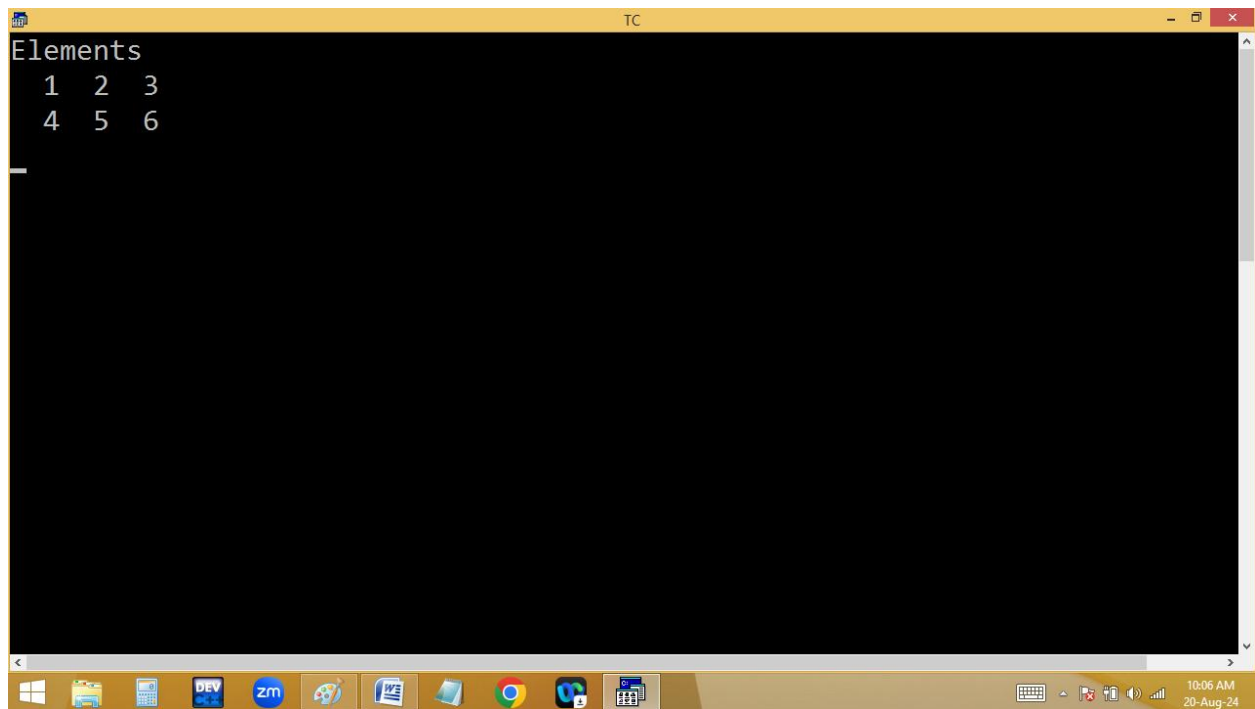
```
TC
Error: Size of structure or array not known in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



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 bottom indicates 'Line 2', 'Col 1', and 'Insert Indent Tab Fill Unindent * E:9AM.C'. The code is as follows:

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

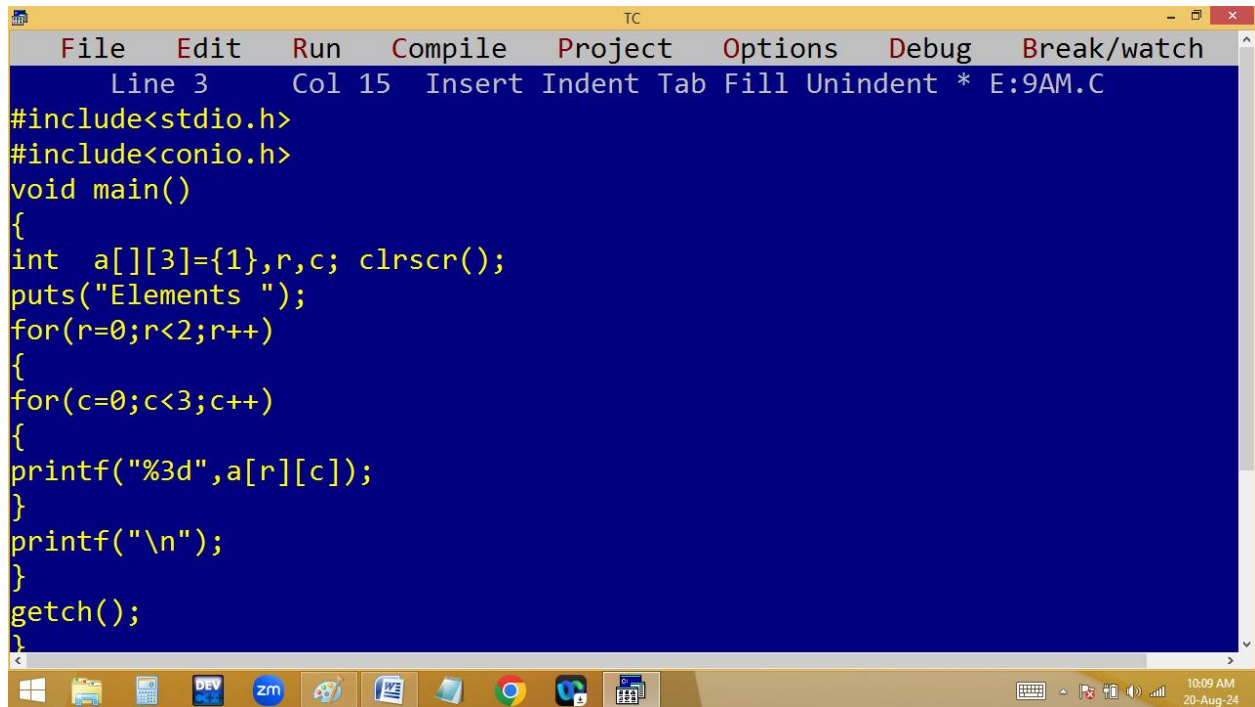
The Windows taskbar at the bottom shows various icons including the Start button, File Explorer, Calculator, DEV C++, ZOOM, and several other applications. The system clock shows 10:06 AM on 20-Aug-24.



The screenshot shows the Turbo C++ (TC) IDE with a black background, displaying the output of the program. The text 'Elements' is printed on the first line. Below it, the numbers 1, 2, 3 are printed on the second line, and 4, 5, 6 are printed on the third line, with a blank line following. The status bar at the bottom indicates 'Line 2', 'Col 1', and 'Insert Indent Tab Fill Unindent * E:9AM.C'. The Windows taskbar at the bottom shows various icons including the Start button, File Explorer, Calculator, DEV C++, ZOOM, and several other applications. The system clock shows 10:06 AM on 20-Aug-24.

```
TC
Line 2   Col 1   Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[][3]={1,2,3},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```

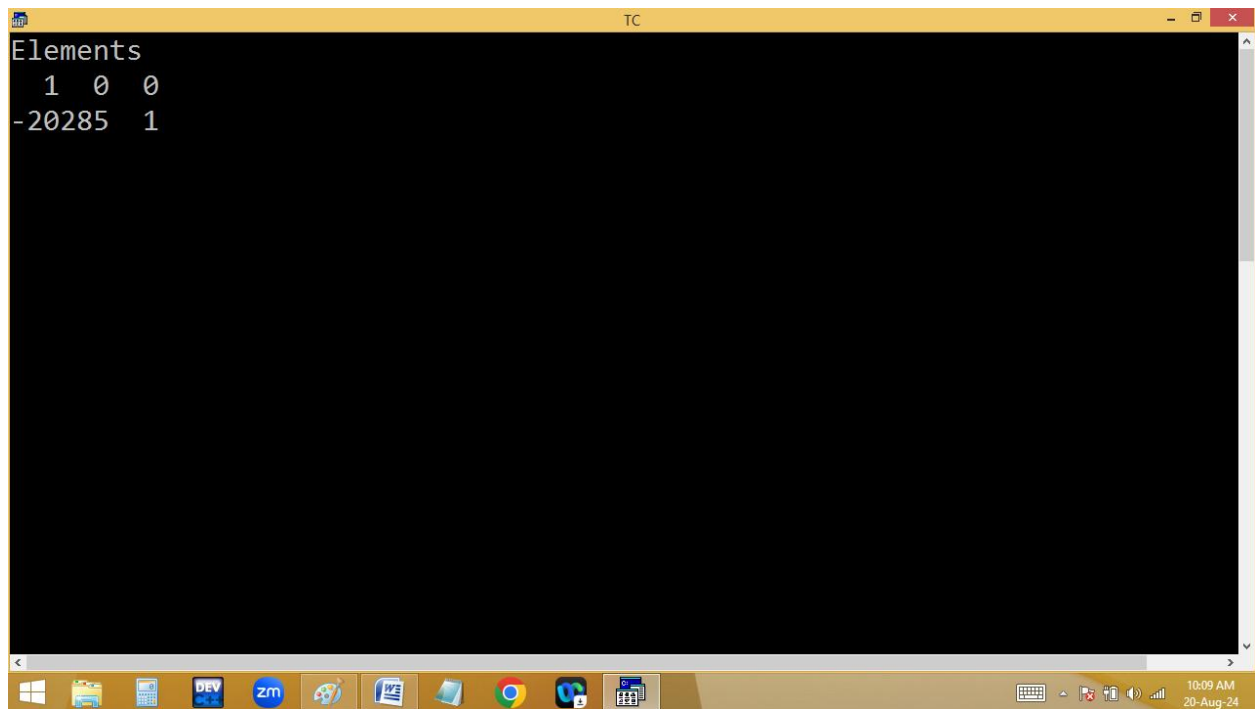
```
TC
Elements
  1  2  3
-20285  1
```



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 3 Col 15 Insert Indent Tab Fill Unindent * E:9AM.C'. The code is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int  a[][3]={1},r,c; clrscr();
    puts("Elements ");
    for(r=0;r<2;r++)
    {
        for(c=0;c<3;c++)
        {
            printf("%3d",a[r][c]);
        }
        printf("\n");
    }
    getch();
}
```

The Windows taskbar at the bottom shows various application icons and the system clock indicating 10:09 AM on 20-Aug-24.



The screenshot shows the Turbo C++ (TC) IDE with a black background, displaying the output of the program. The output is as follows:

```
Elements
  1  0  0
-20285  1
```

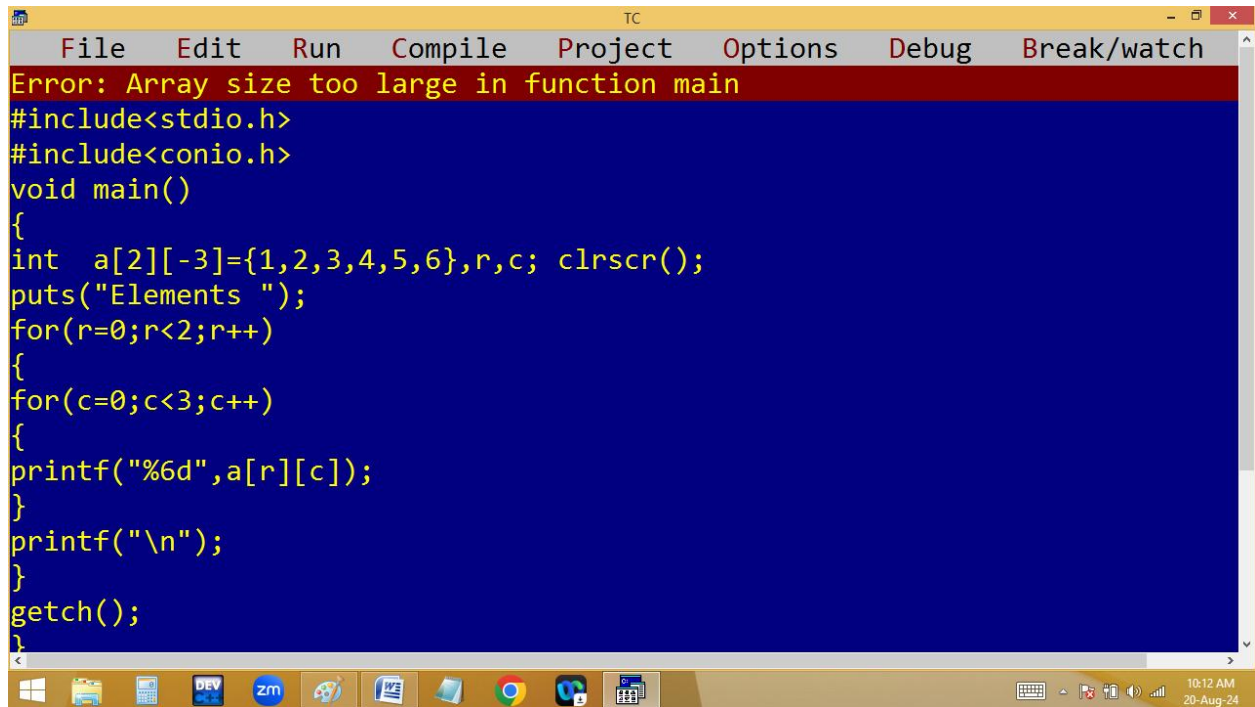
The Windows taskbar at the bottom shows various application icons and the system clock indicating 10:09 AM on 20-Aug-24.

```
TC
Line 17 Col 22 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={9},r,c; clrscr();
a[0][0]=1; a[1][2]=6;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```

```
TC
Elements
 1  0  0
 0  0  6
```

```
TC
Line 12 Col 11 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3],r,c; clrscr();
a[0][0]=1; a[1][2]=6;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

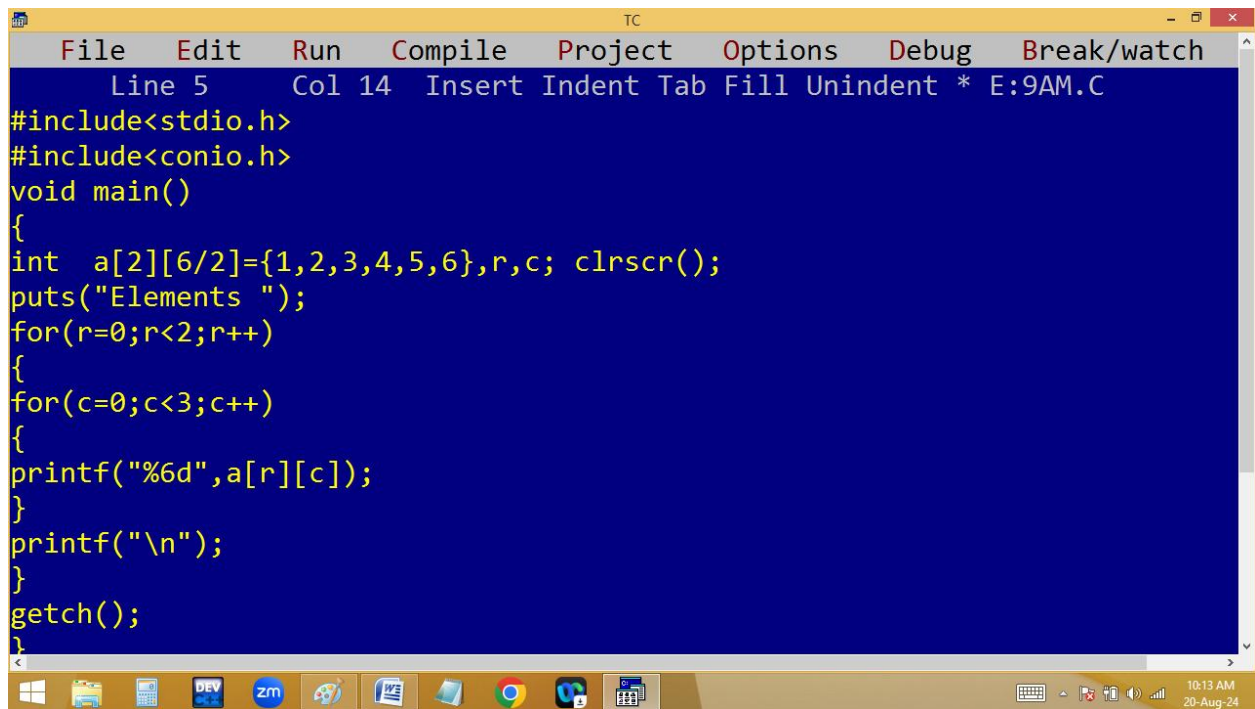
```
TC
Elements
    1  7083  1824
   -20 5201    6
```



The screenshot shows the Turbo C++ (TC) IDE with a red error message at the top: "Error: Array size too large in function main". The code in the editor is as follows:

```
File Edit Run Compile Project Options Debug Break/watch
Error: Array size too large in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][-3]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

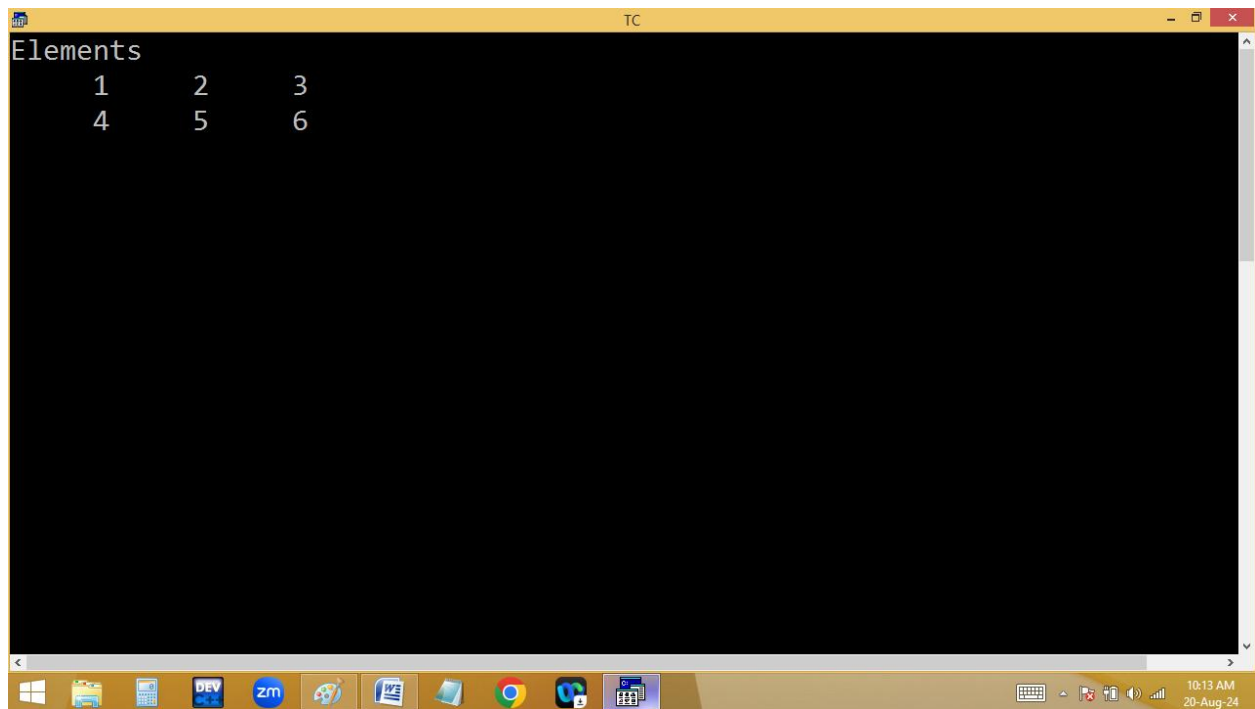
The Windows taskbar at the bottom shows the time as 10:12 AM on 20-Aug-24.



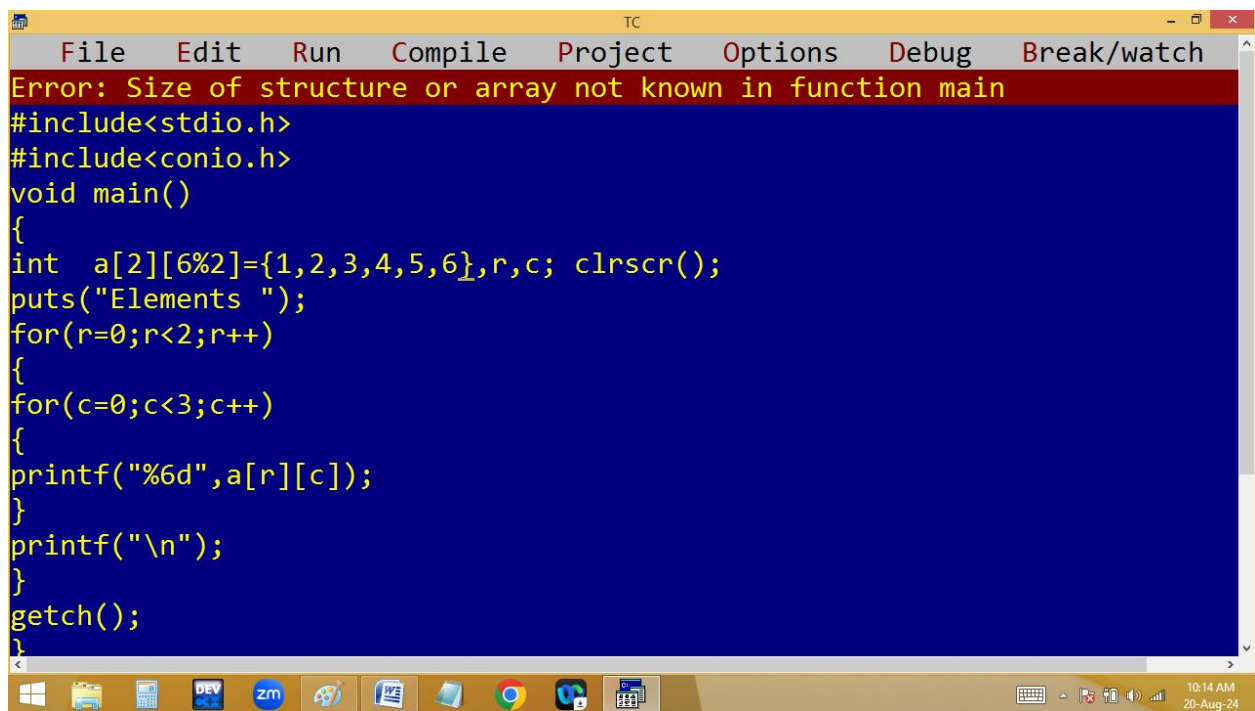
The screenshot shows the Turbo C++ (TC) IDE with the same code as the first image, but the error message has been removed. The code is now:

```
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 14 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][6/2]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

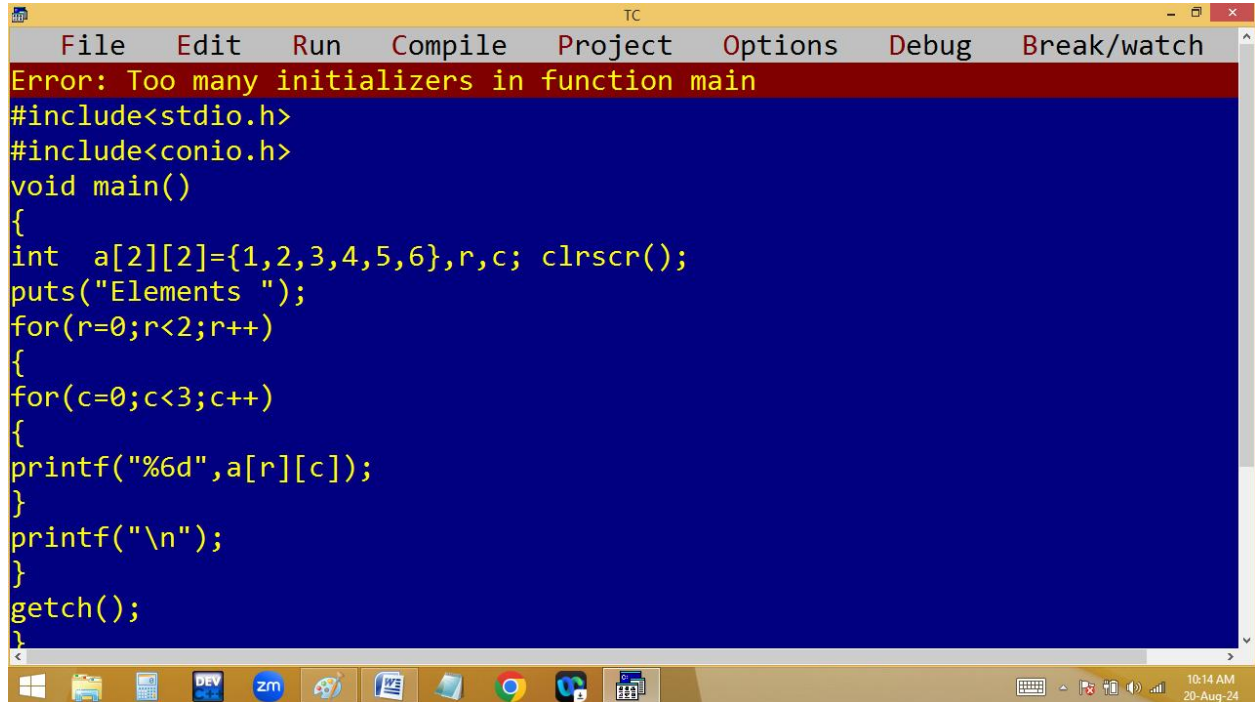
The Windows taskbar at the bottom shows the time as 10:13 AM on 20-Aug-24.



```
Elements
  1    2    3
  4    5    6
```



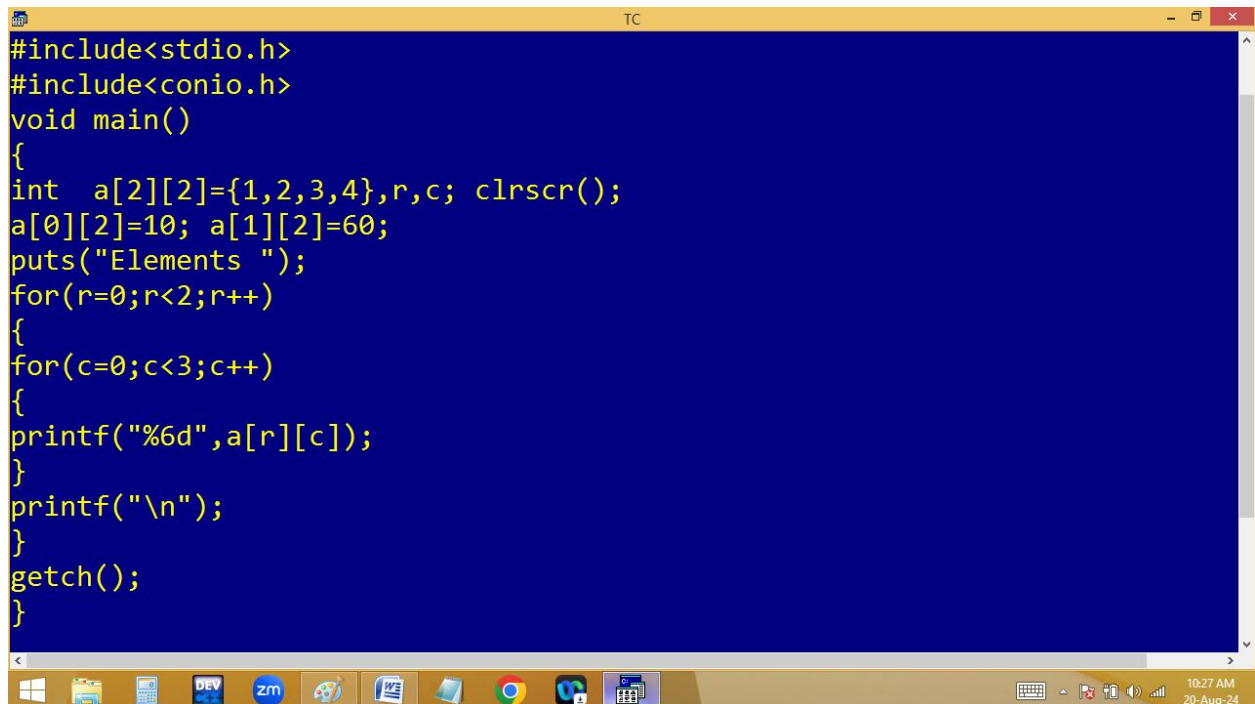
```
File Edit Run Compile Project Options Debug Break/watch
Error: Size of structure or array not known in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][6%2]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

The screenshot shows the Turbo C++ (TC) IDE with a red error message at the top: "Error: Too many initializers in function main". The code in the editor is as follows:

```
File Edit Run Compile Project Options Debug Break/watch
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][2]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

The taskbar at the bottom shows various application icons and the system clock indicating 10:14 AM on 20-Aug-24.

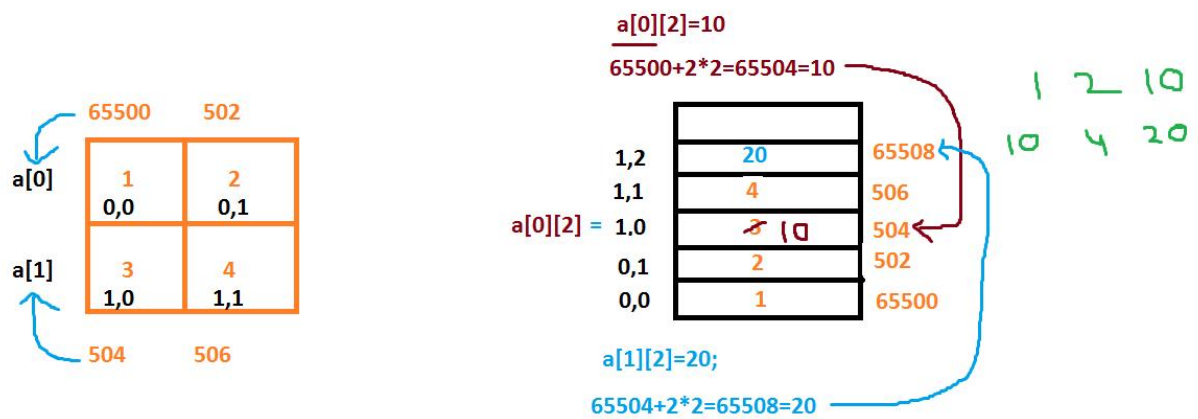


The screenshot shows the Turbo C++ (TC) IDE with the following code in the editor:

```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][2]={1,2,3,4},r,c; clrscr();
a[0][2]=10; a[1][2]=60;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

The taskbar at the bottom shows various application icons and the system clock indicating 10:27 AM on 20-Aug-24.

```
TC
Elements
1      2      10
10     4      60
```



Reading and printing elements of a 2d array:

```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("Elements ");
for(r=0;r<nr;r++)
{
for(c=0;c<nc;c++)
{printf("%6d",a[r][c]);}
printf("\n");
}
getch();
}
```

```
TC
Enter no of rows and columns 2 3
Enter 6 integers
1 2 3 4 5 6
Elements
    1      2      3
    4      5      6
```

```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("Elements ");
for(r=0;r<nr;r++)
{
for(c=0;c<nc;c++)
{printf("%6d",a[r][c]);}
printf("\n");
}
getch();
}
```

```
TC
Enter no of rows and columns 3 3
Enter 9 integers
2 0 4 8 1 6 4 0 8
Elements
      2      0      4
      8      1      6
      4      0      8
```

Read n elements to n*n array and find the no of even, odd, zero elements row wise.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c,e,o,z; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("\t Even\tOdd\tZero");
puts("-----");
for(r=0;r<nr;r++)
{for(e=o=z=c=0;c<nc;c++)
{if(a[r][c]==0)z++; else if(a[r][c]%2==0)e++; else o++;}
printf("%d-row\t %d\t%d\t%d\n",r+1,e,o,z);
}
getch();
}
```

```
Enter no of rows and columns 3 4
Enter 12 integers
1 2 0 3
2 4 5 7
0 0 1 4
      Even   Odd   Zero
-----
1-row   1     2     1
2-row   2     2     0
3-row   1     1     2
```

```

puts("\t Even\tOdd\tZero");
puts("-----");
for( r=0;r<3;r++ )
{
for( e=o=z=c=0;c<4;c++)
{
if(a[r][c]==0)z++;else if(a[r][c]%2==0)e++;else o++;
}
printf("%d-row\t%d\t%d\t%d\n",r+1, e, o, z);
}

```

1	0	2	3
4	1	0	0
3	7	2	8

	Even	Odd	Zero
1-row	1	2	1
2-row	1	1	2
3-row	2	2	0

Column wise: