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.

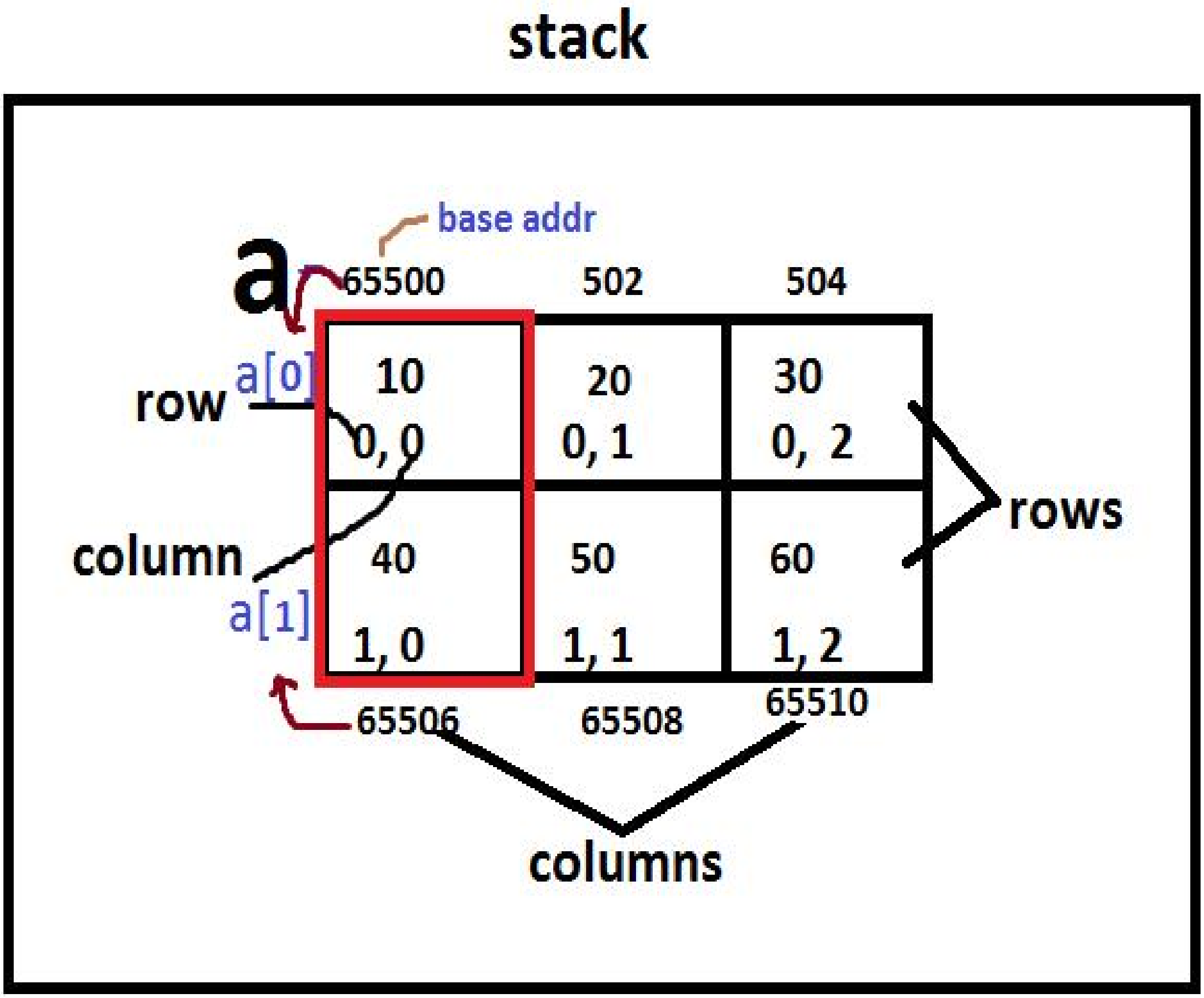
It is a n*n matrix.

Syntax:

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

Eg:

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



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

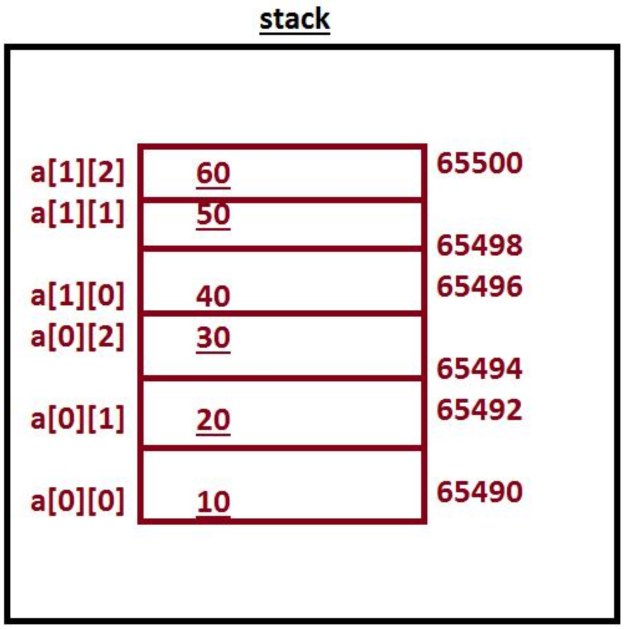
Internally how this statement is working ?

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

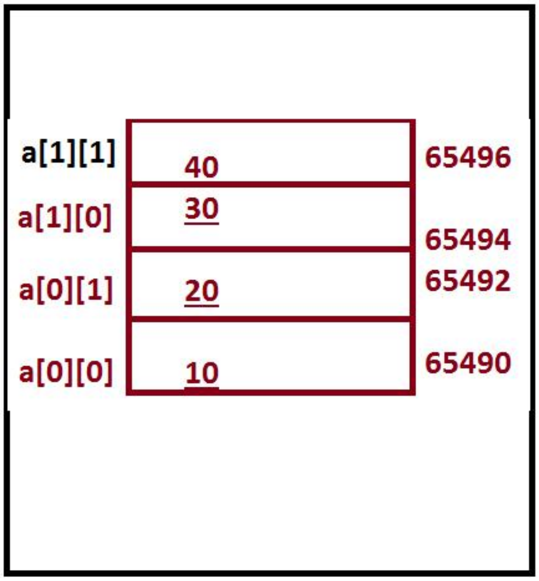
Index no

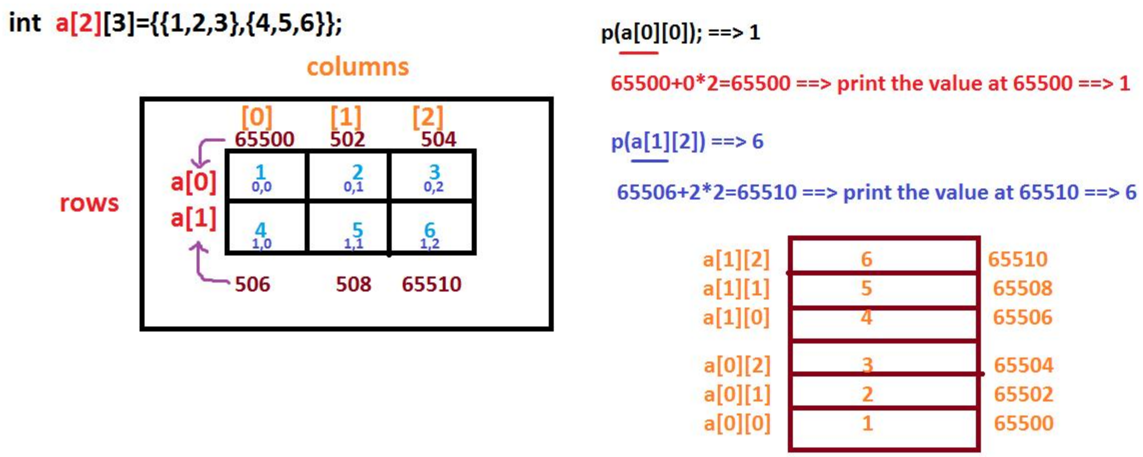
65500 + [0] col → 65500 + 0 * 2 → 65500 → value at 65500 is 10.

Int size



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

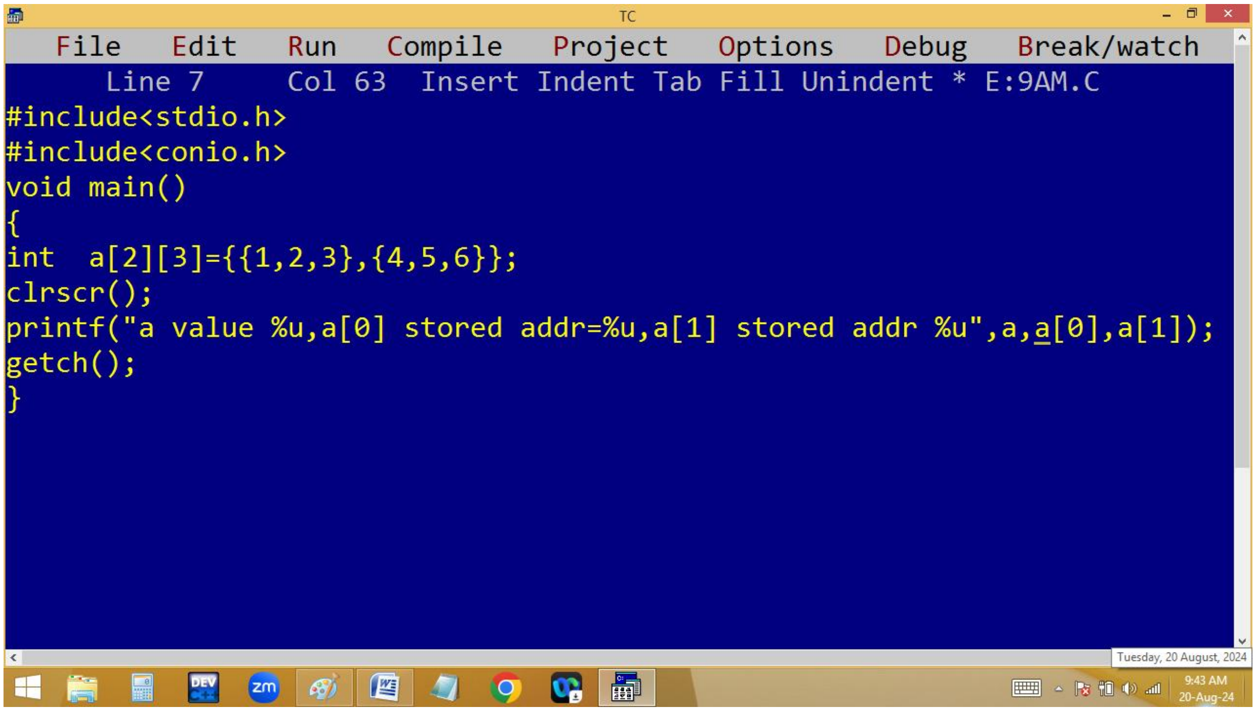




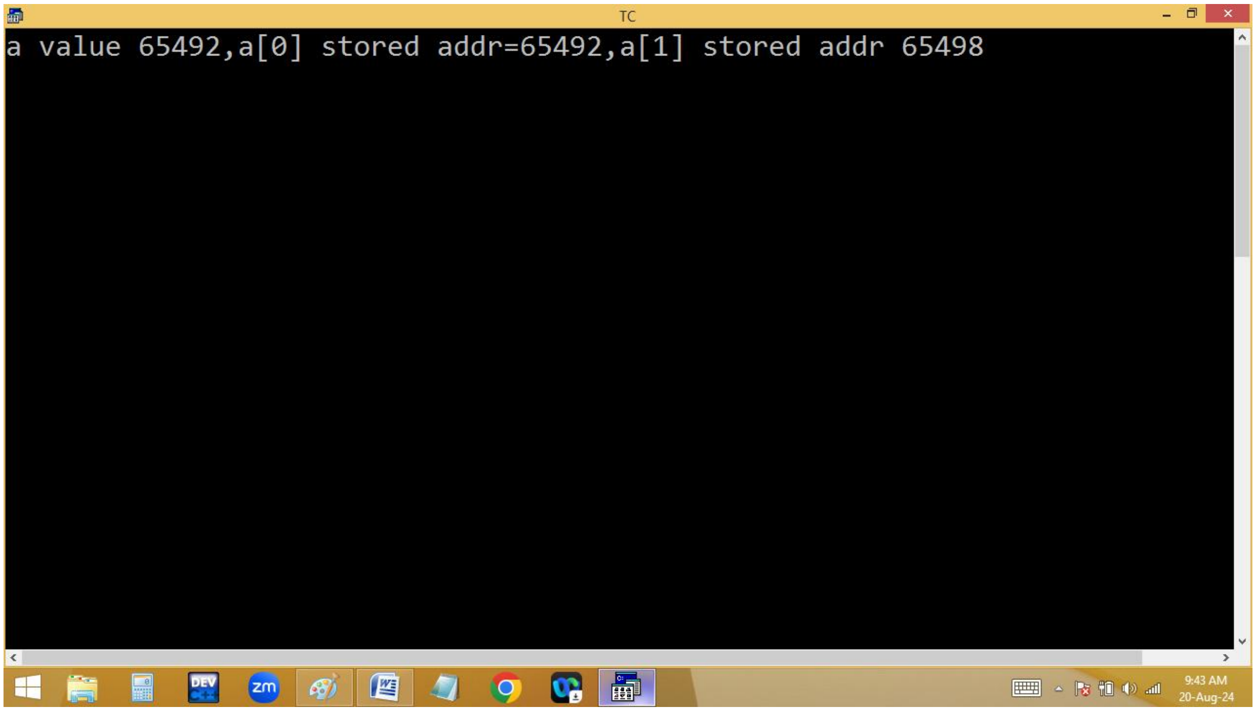
```
int  a[2][3]={{1,2,3},{4,5,6}};
```

```
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
```

Finding array address:

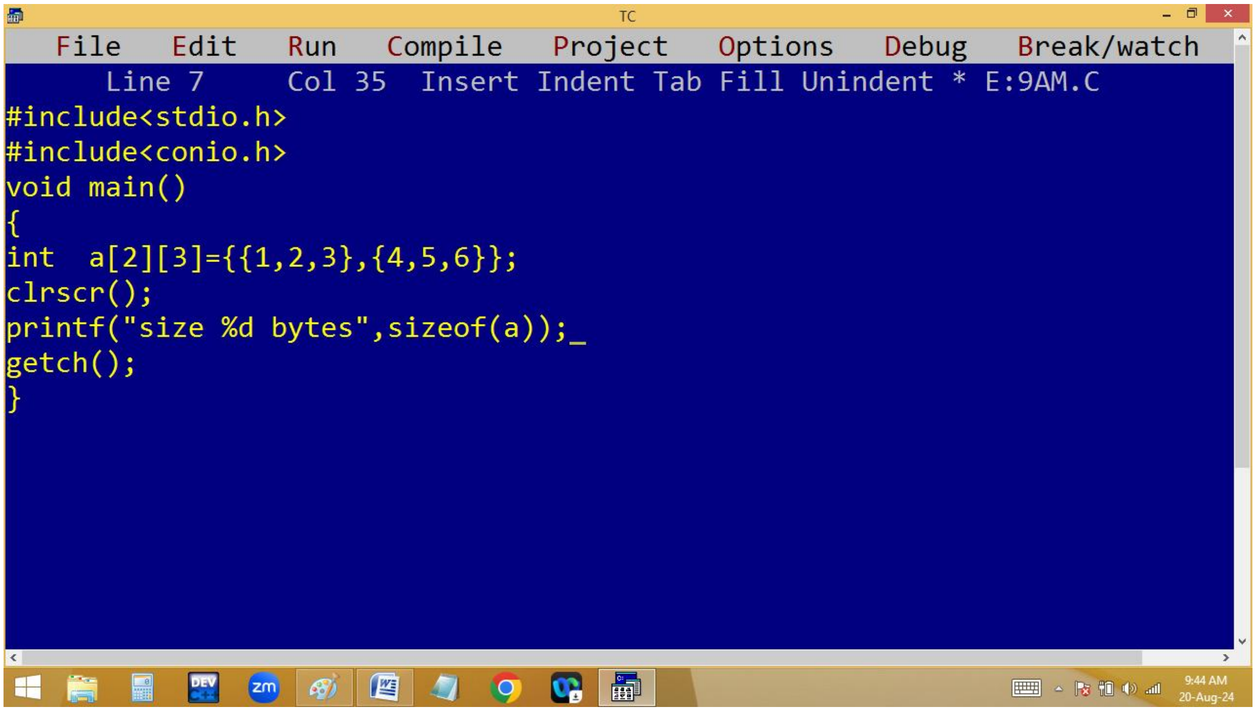

TC
File Edit Run Compile Project Options Debug Break/watch
Line 7 Col 63 Insert Indent Tab Fill Unindent * E:9AM.C
Tuesday, 20 August, 2024
9:43 AM
20-Aug-24

```
#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();
}
```

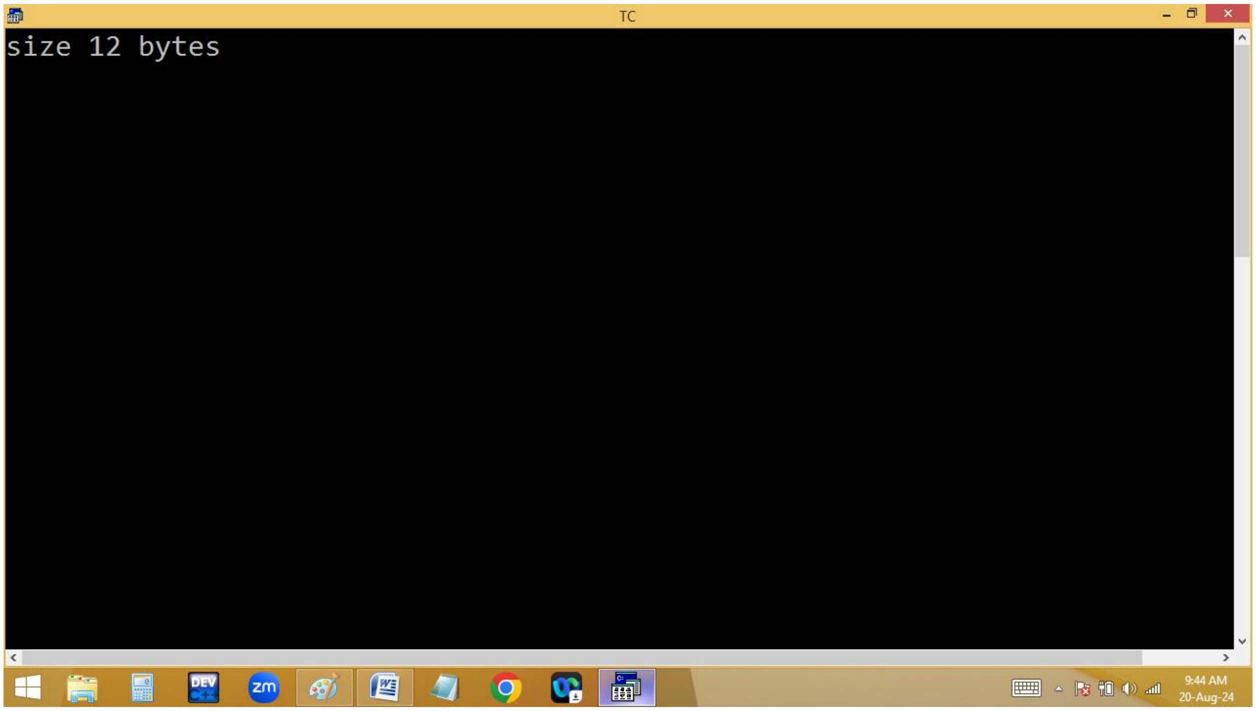


```
a value 65492,a[0] stored addr=65492,a[1] stored addr 65498
```

Finding n*n matrix size:

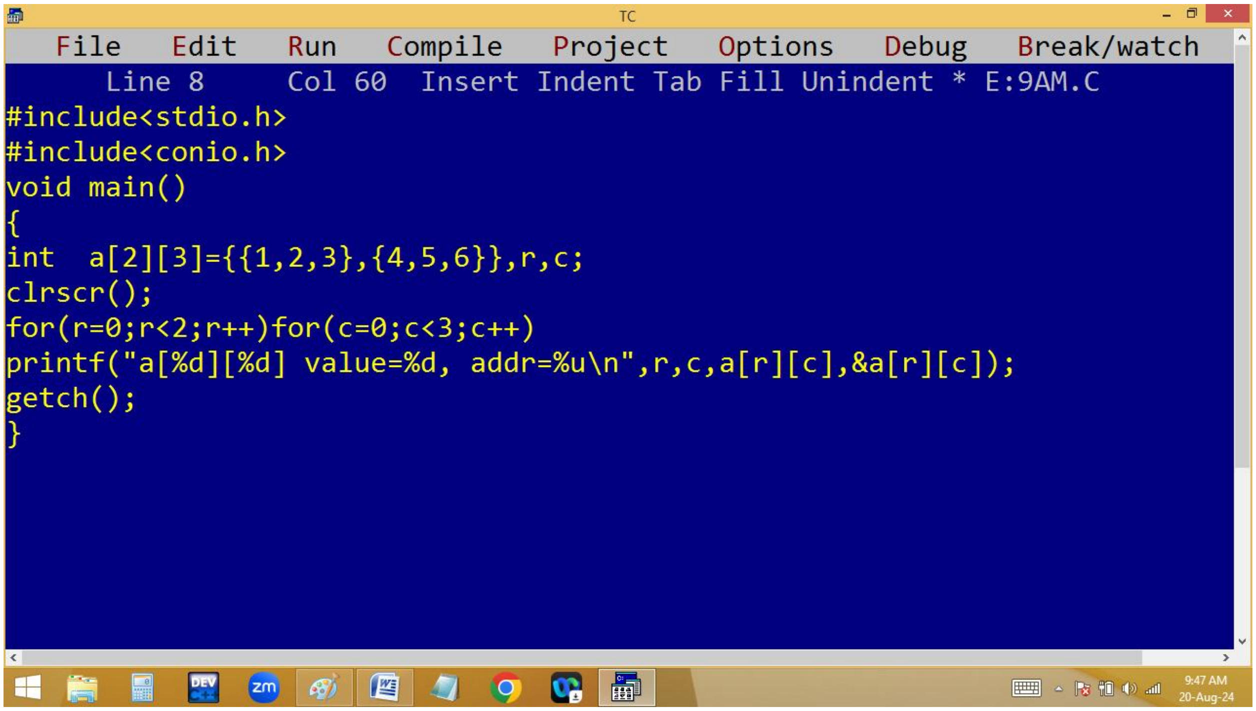

TC
File Edit Run Compile Project Options Debug Break/watch
Line 7 Col 35 Insert Indent Tab Fill Unindent * E:9AM.C
9:44 AM
20-Aug-24

```
#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();
}
```

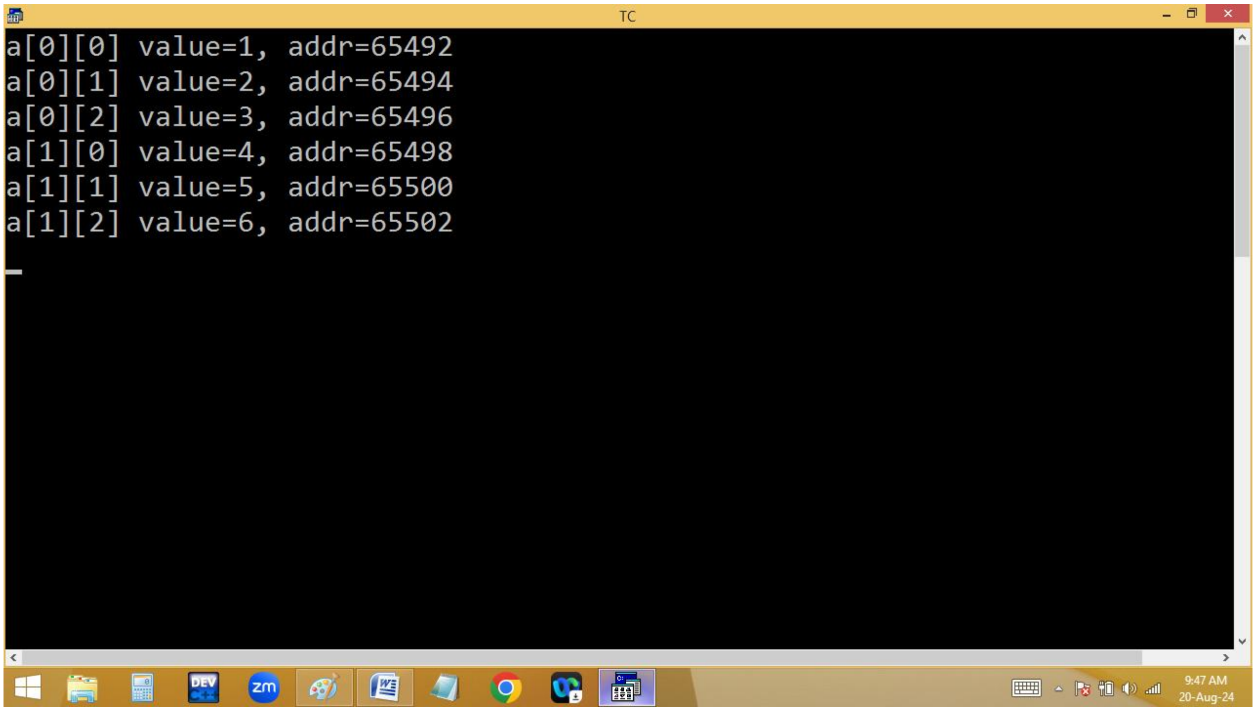


```
size 12 bytes
```

Finding array value, address and position:

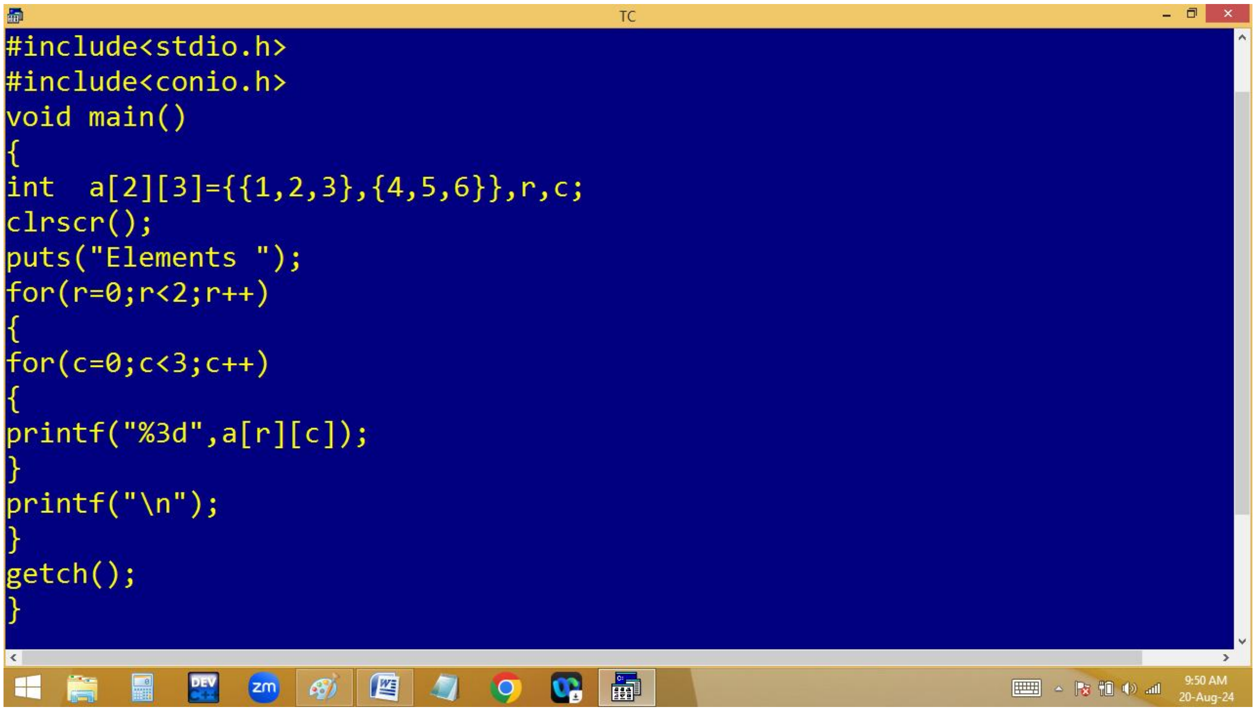


```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2,3},{4,5,6}},r,c;
clrscr();
for(r=0;r<2;r++)for(c=0;c<3;c++)
printf("a[%d][%d] value=%d, addr=%u\n",r,c,a[r][c],&a[r][c]);
getch();
}
```

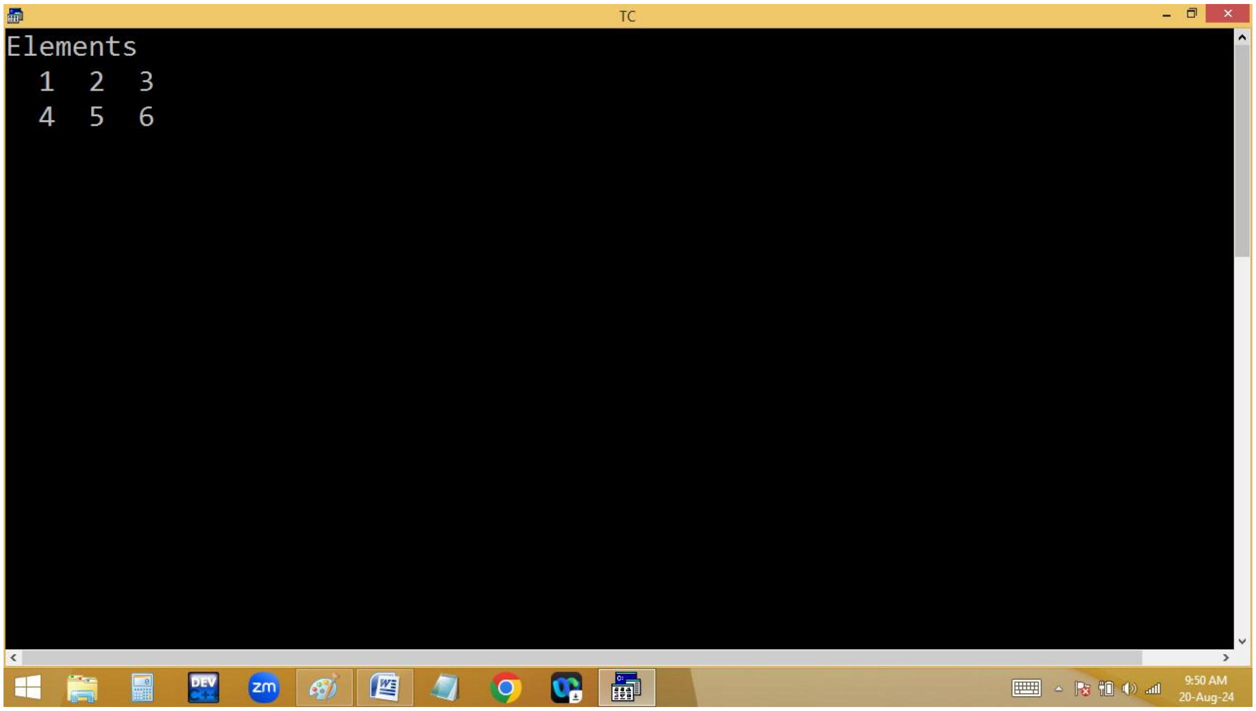


```
a[0][0] value=1, addr=65492
a[0][1] value=2, addr=65494
a[0][2] value=3, addr=65496
a[1][0] value=4, addr=65498
a[1][1] value=5, addr=65500
a[1][2] value=6, addr=65502
```

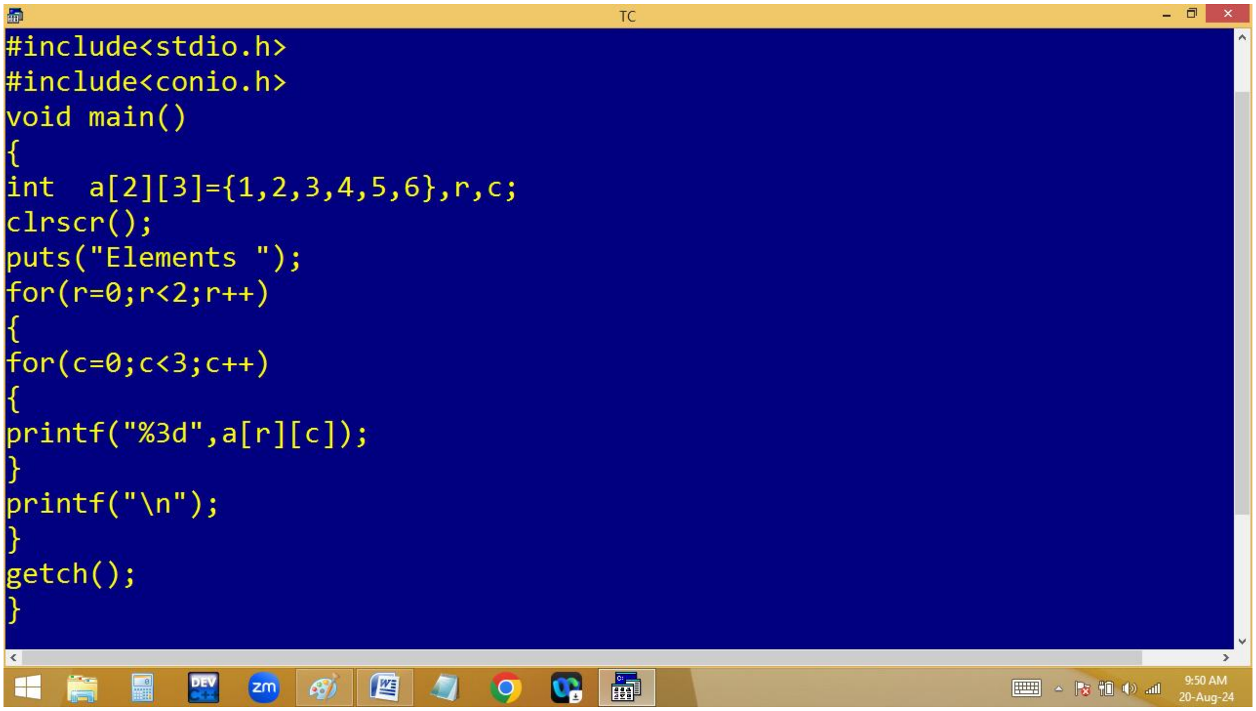
Direct initialization of a n*n matrix:



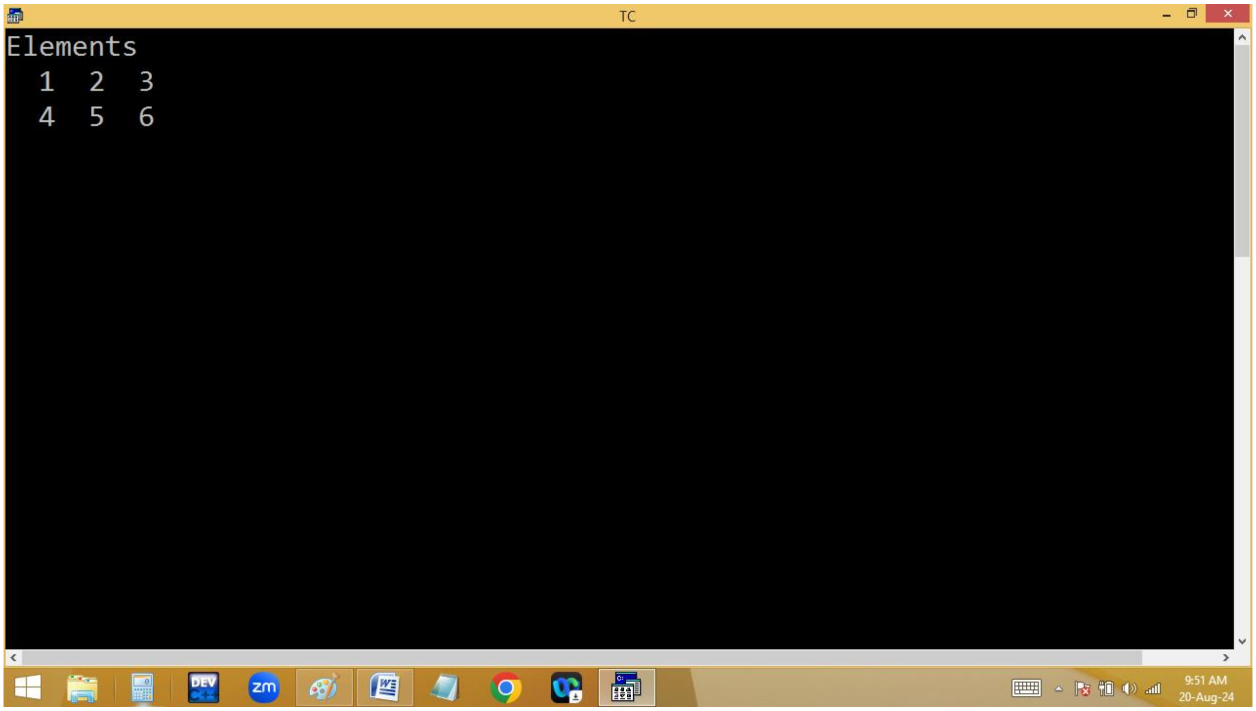
```c
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2,3},{4,5,6}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



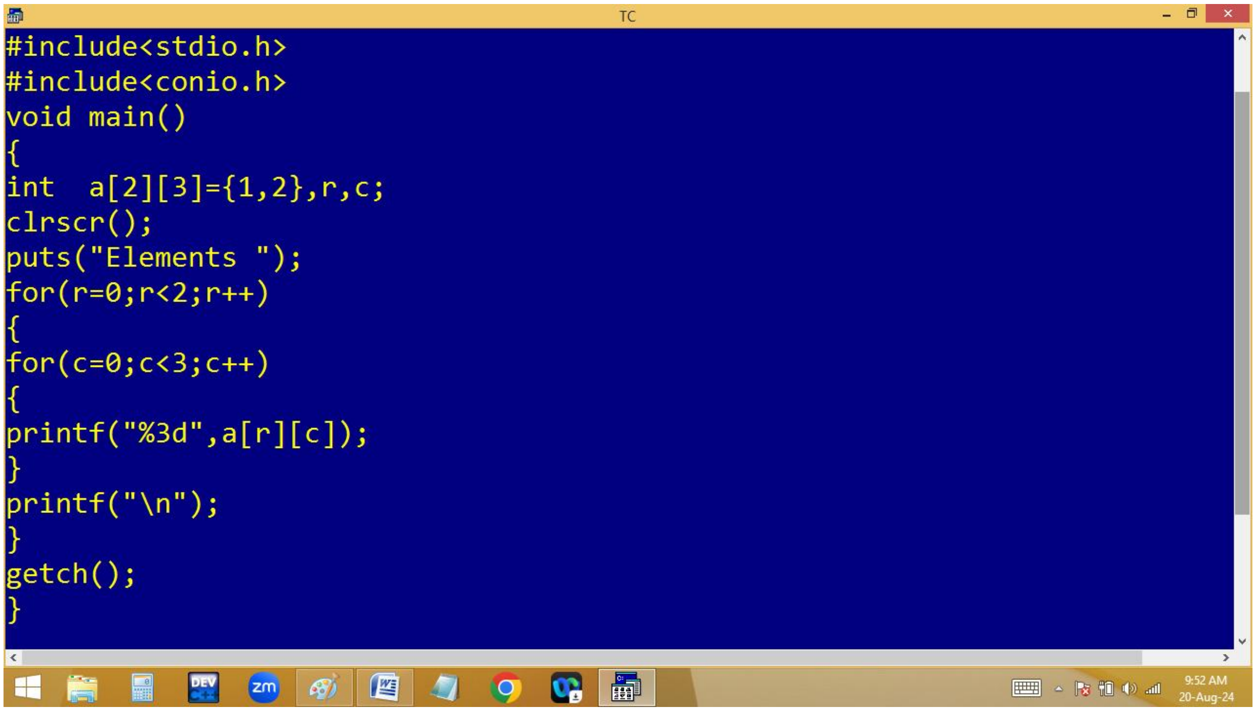
```
Elements
  1  2  3
  4  5  6
```



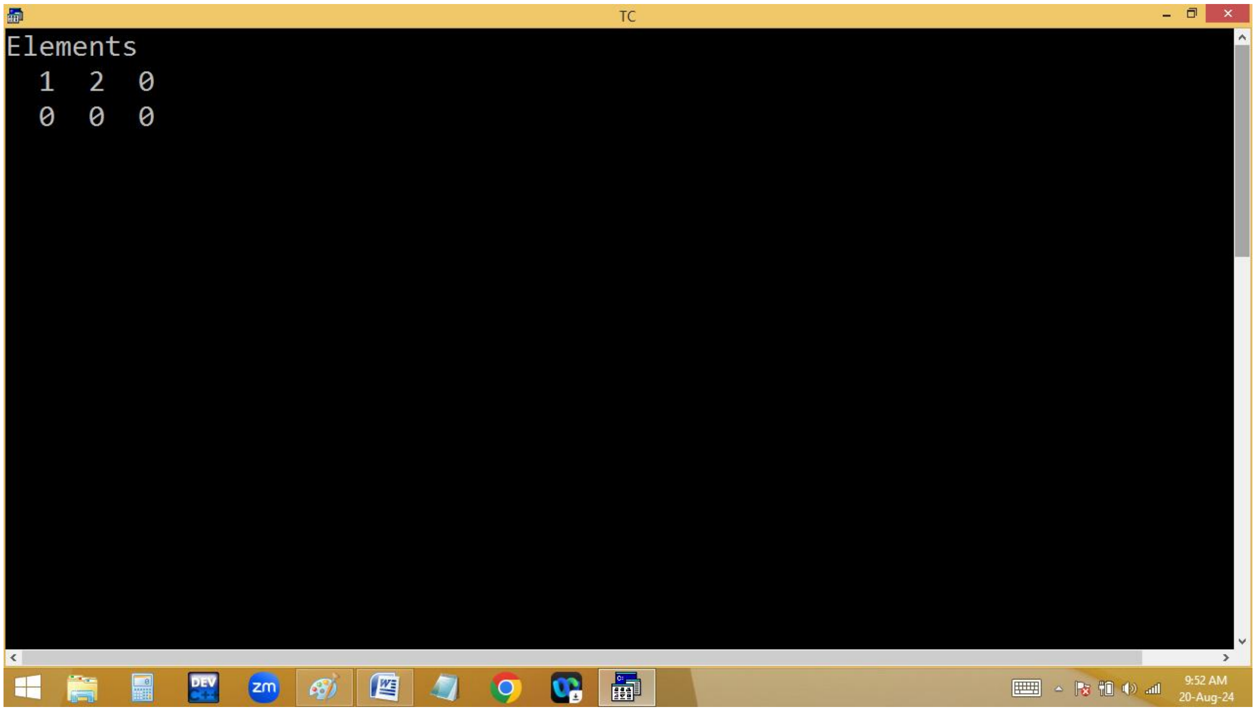
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={1,2,3,4,5,6},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



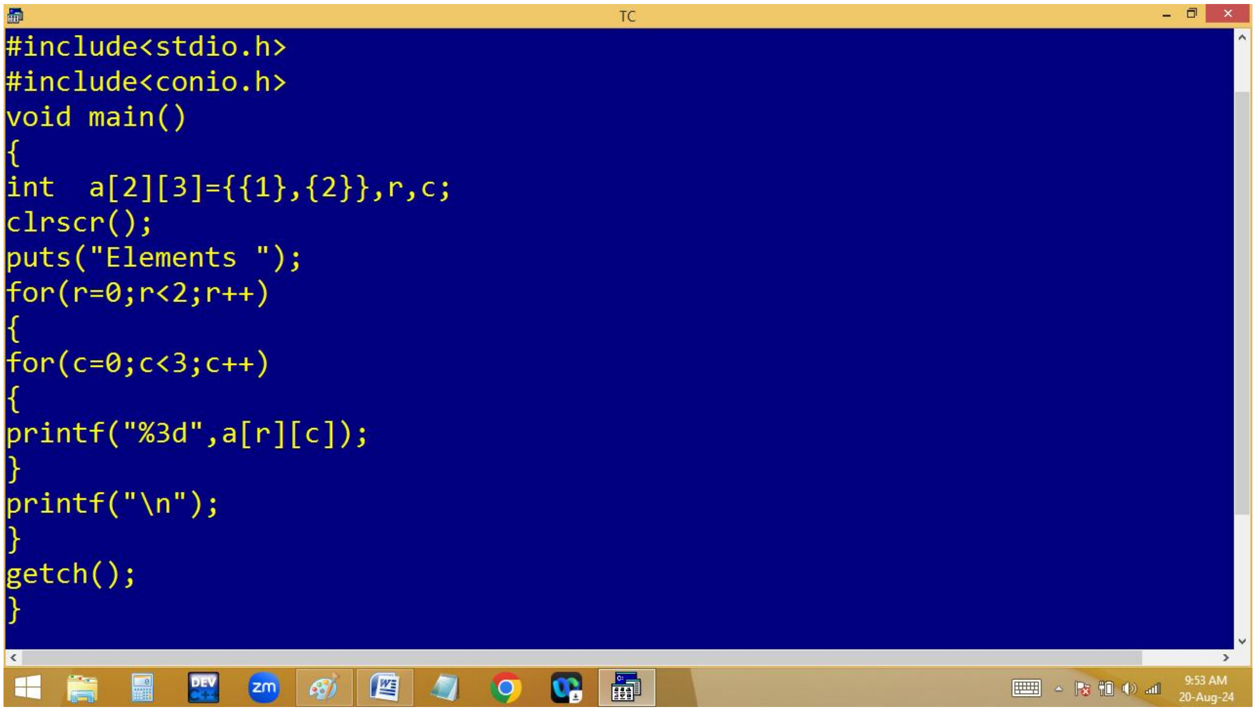
```
Elements
  1  2  3
  4  5  6
```



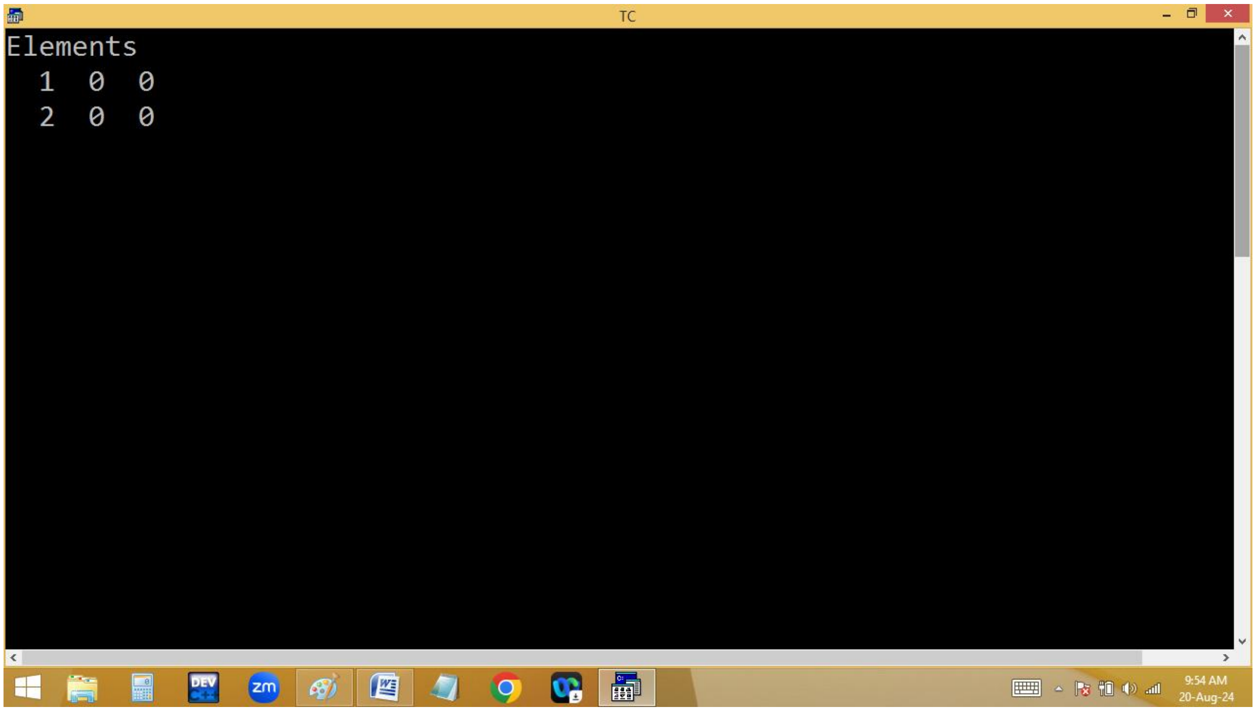
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={1,2},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



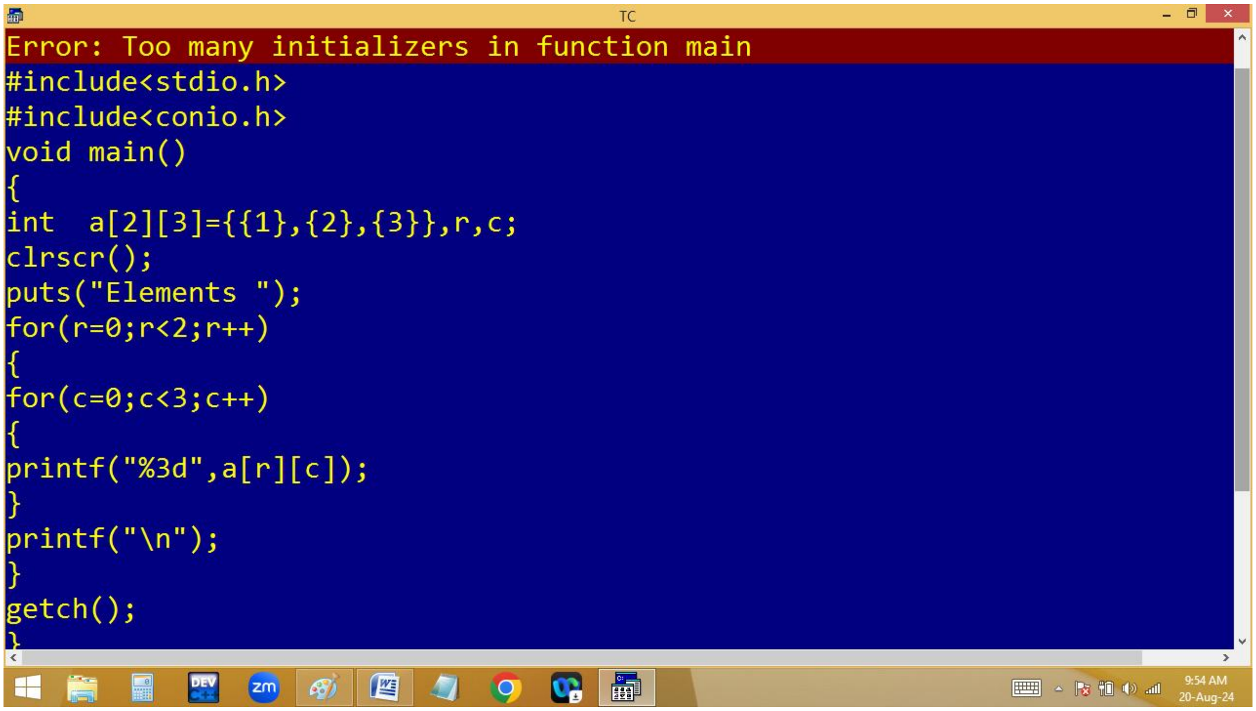
```
Elements
  1  2  0
  0  0  0
```



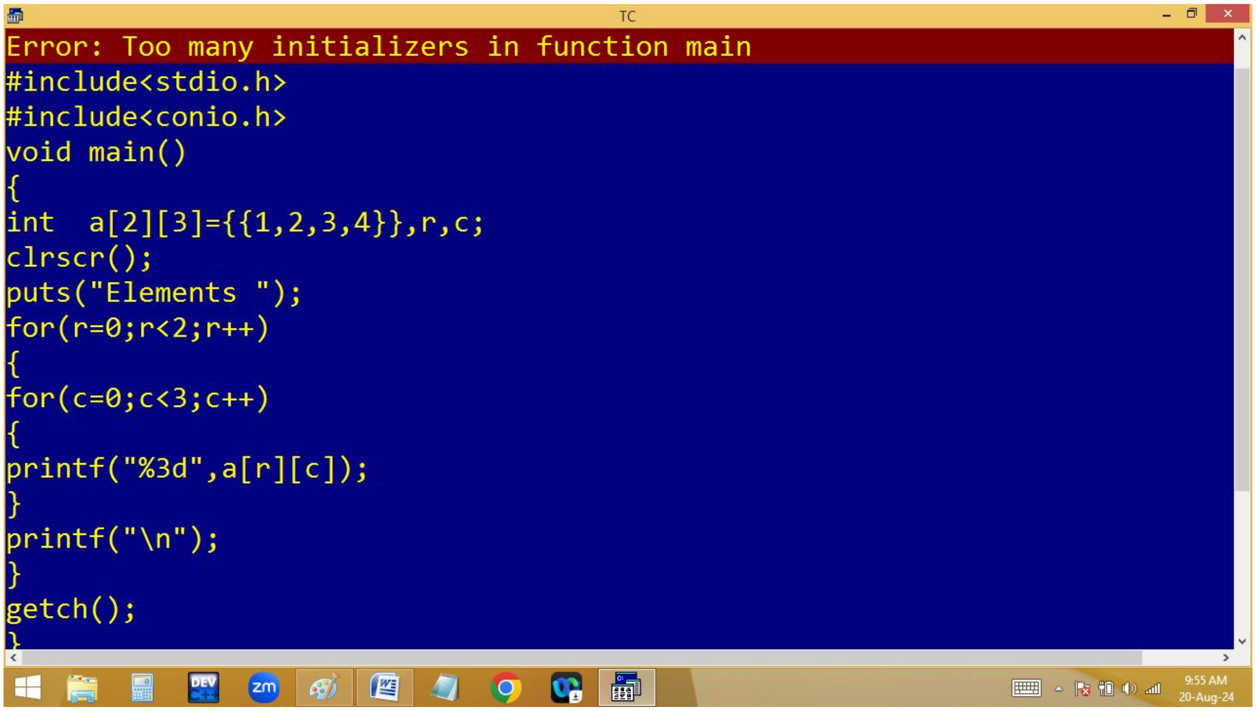
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},{2}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



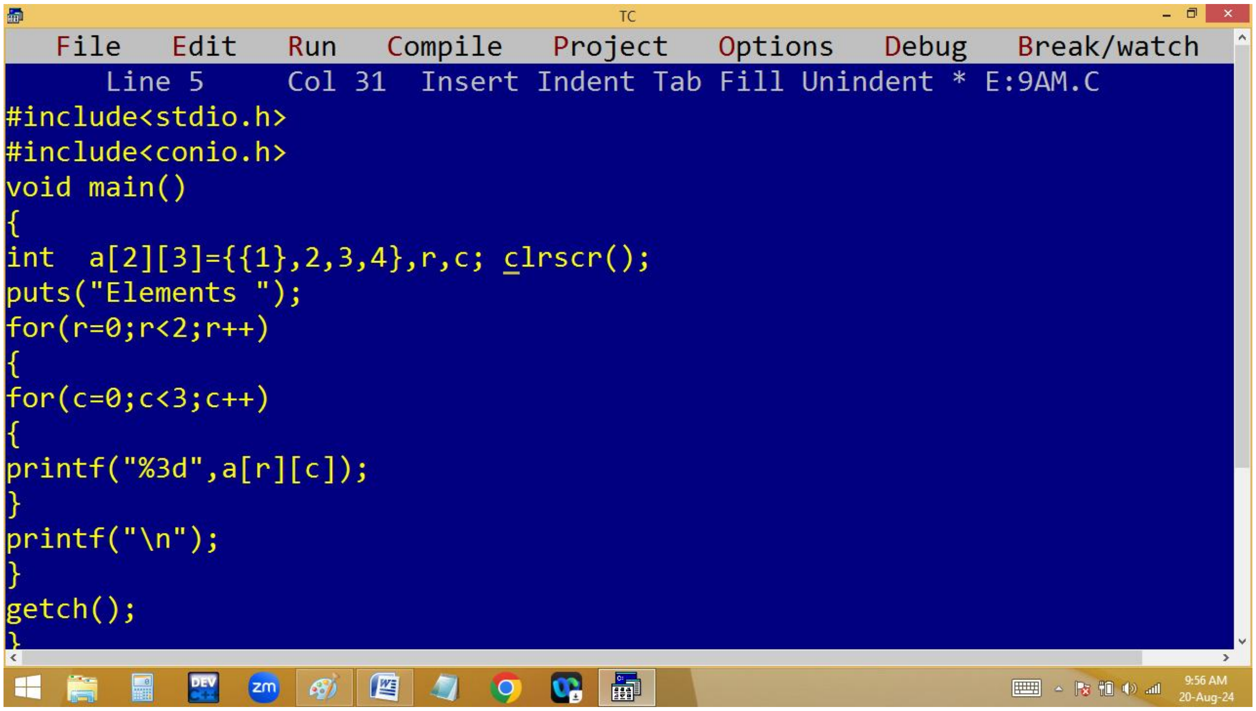
```
Elements
  1  0  0
  2  0  0
```



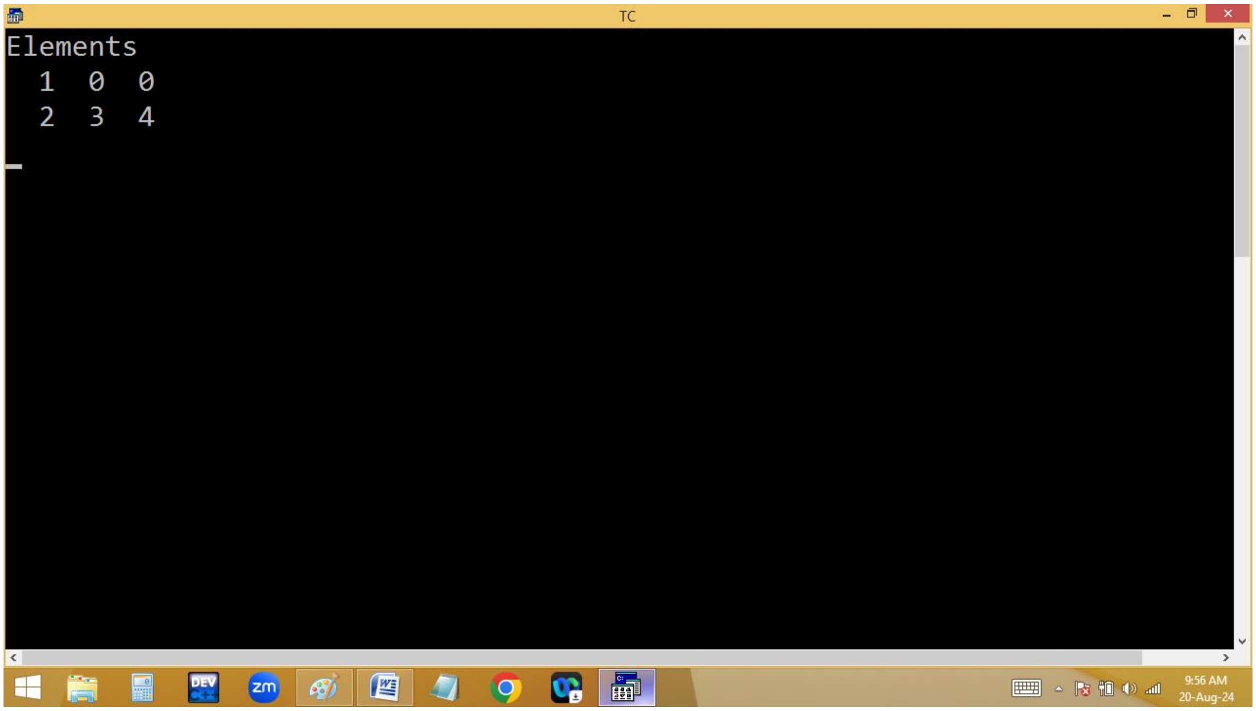
```
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},{2},{3}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



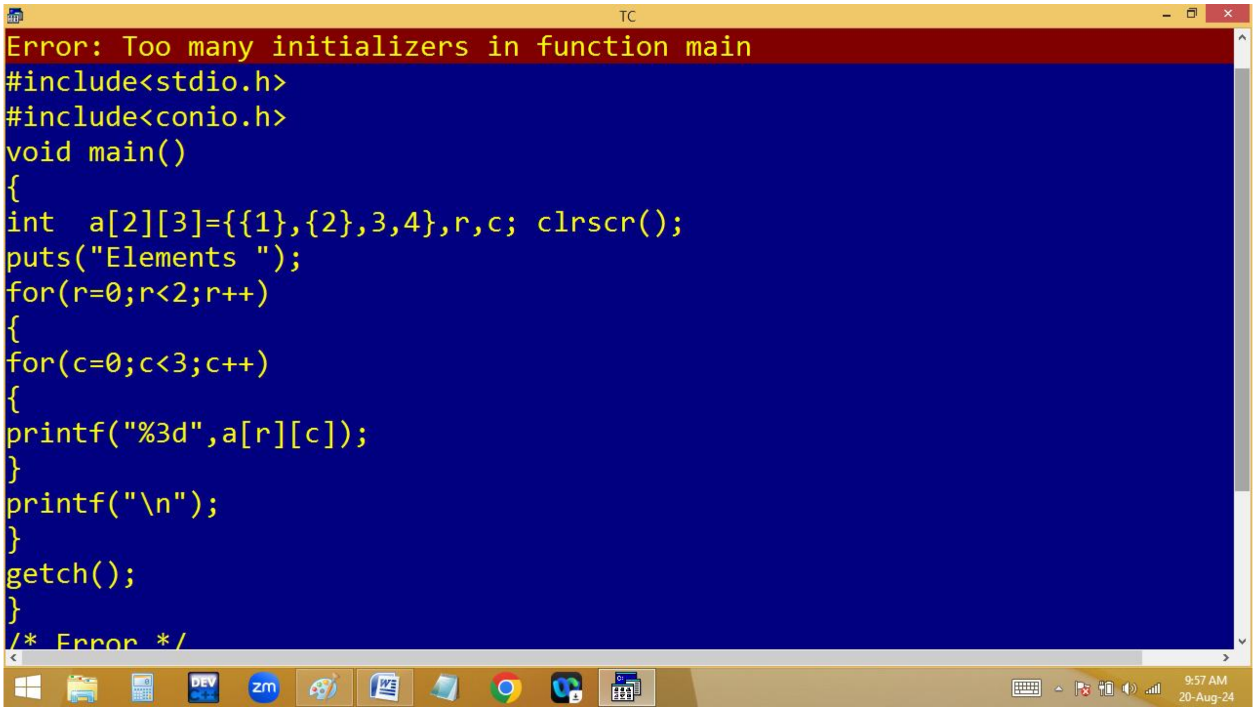
```
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2,3,4}},r,c;
clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```


TC
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 31 Insert Indent Tab Fill Unindent * E:9AM.C
9:56 AM
20-Aug-24

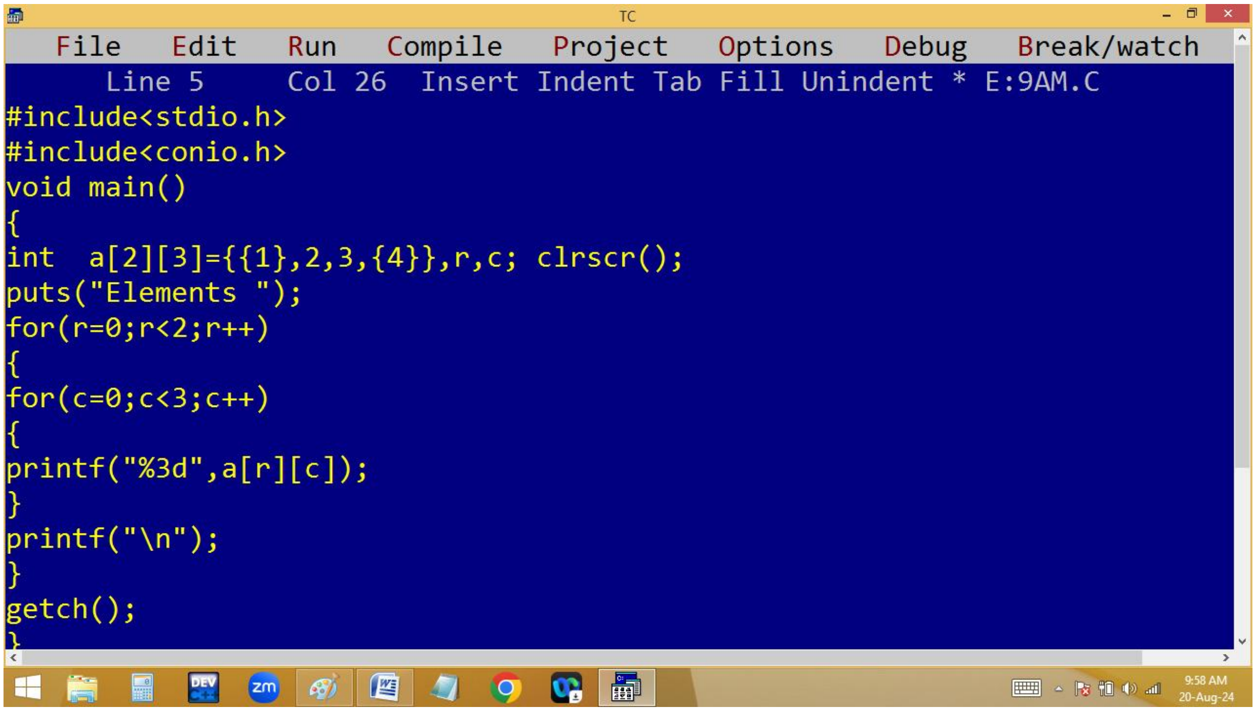
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},2,3,4},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



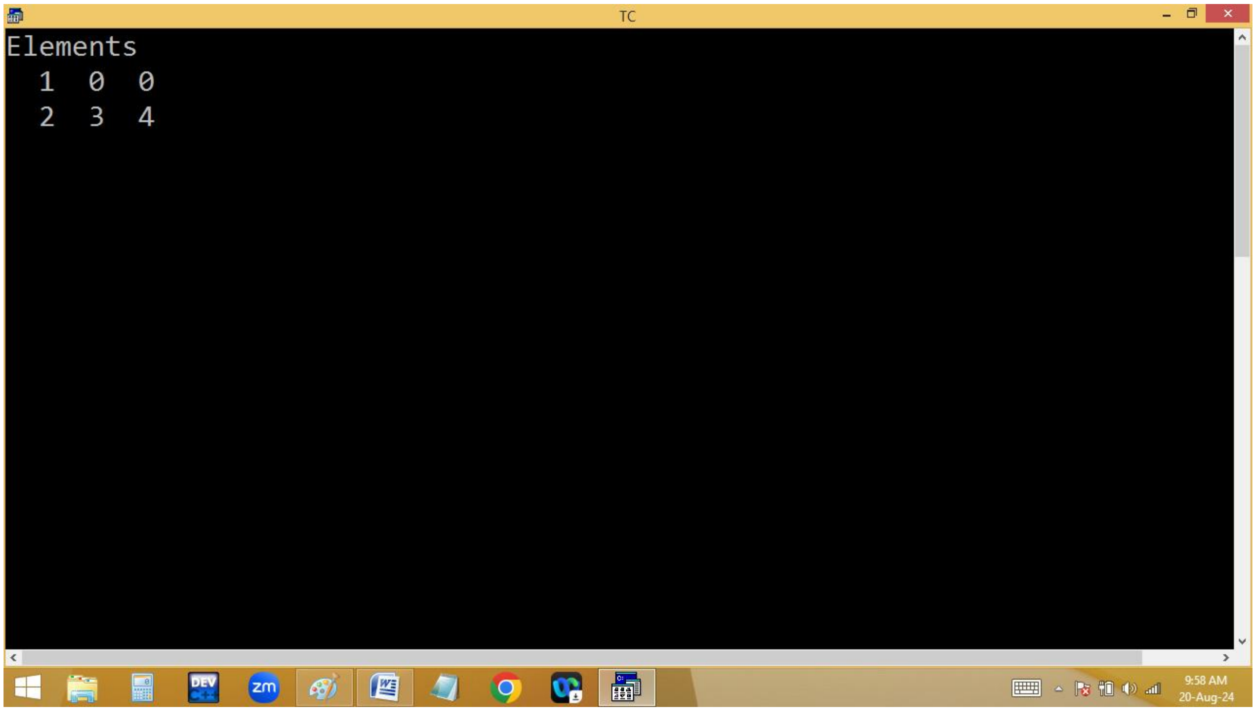
```
Elements
  1  0  0
  2  3  4
```



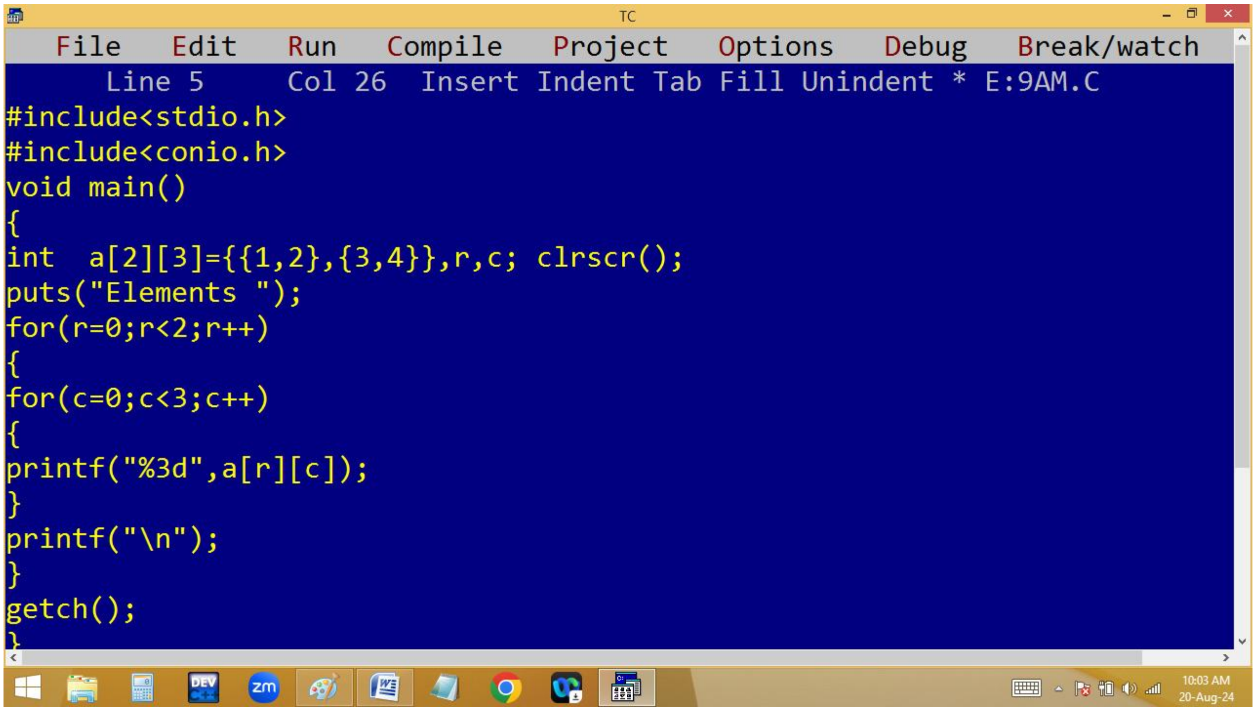
```
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},{2},3,4},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



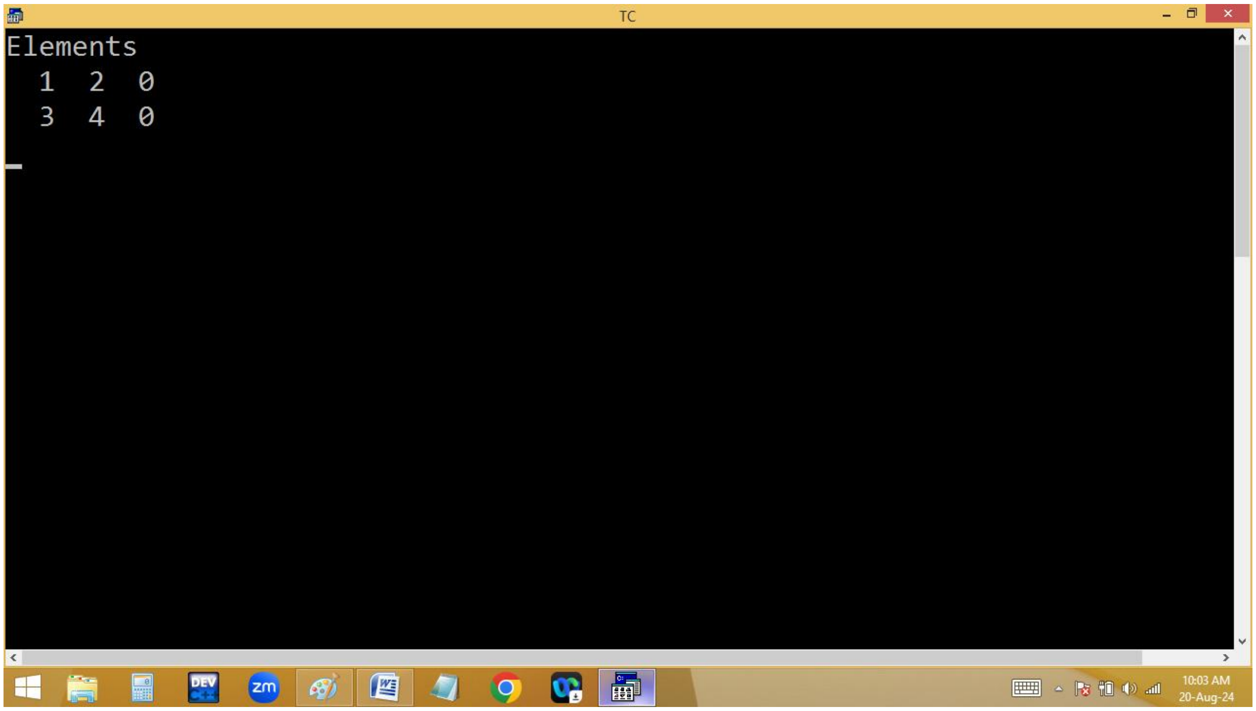
```
      Line 5      Col 26  Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1},2,3,{4}},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



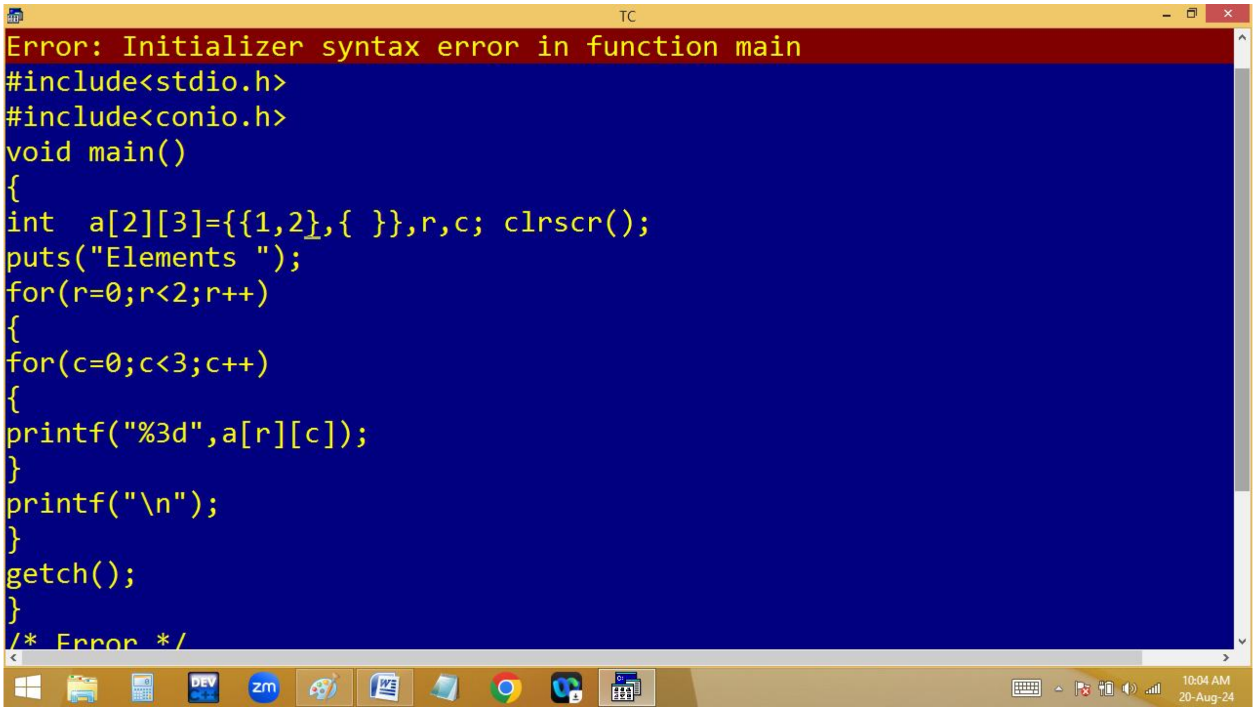
```
Elements
  1  0  0
  2  3  4
```



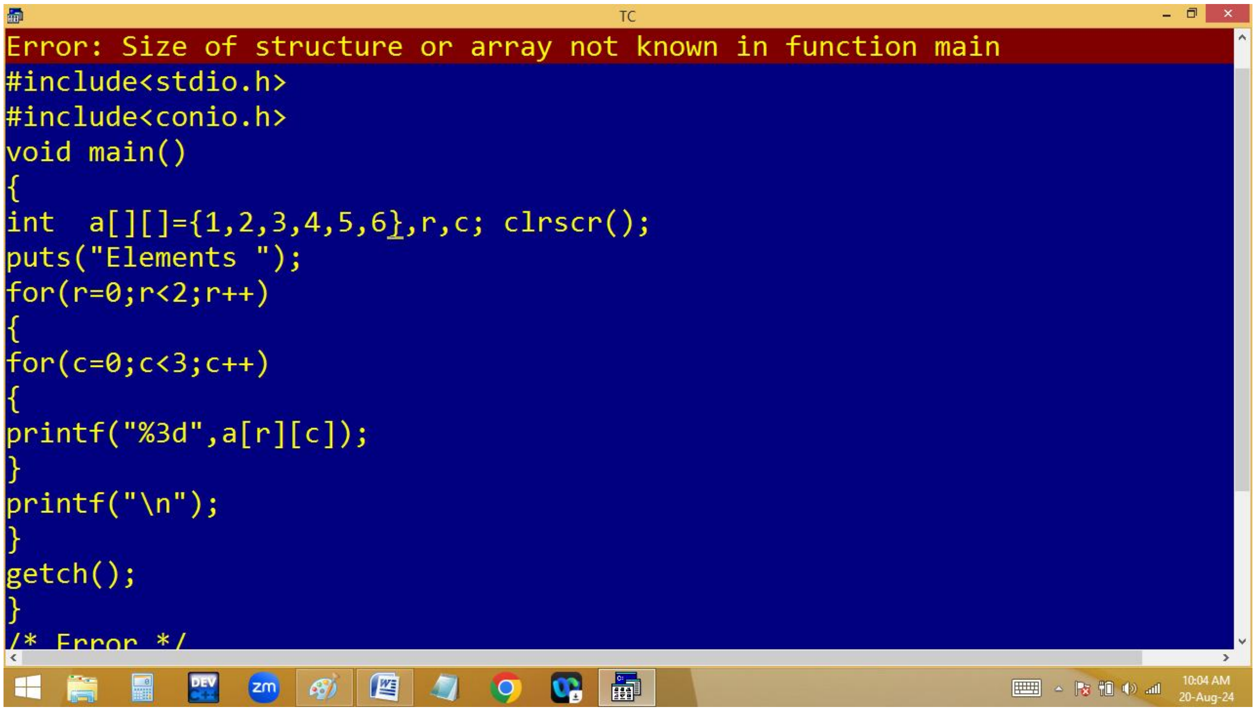
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2},{3,4}},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



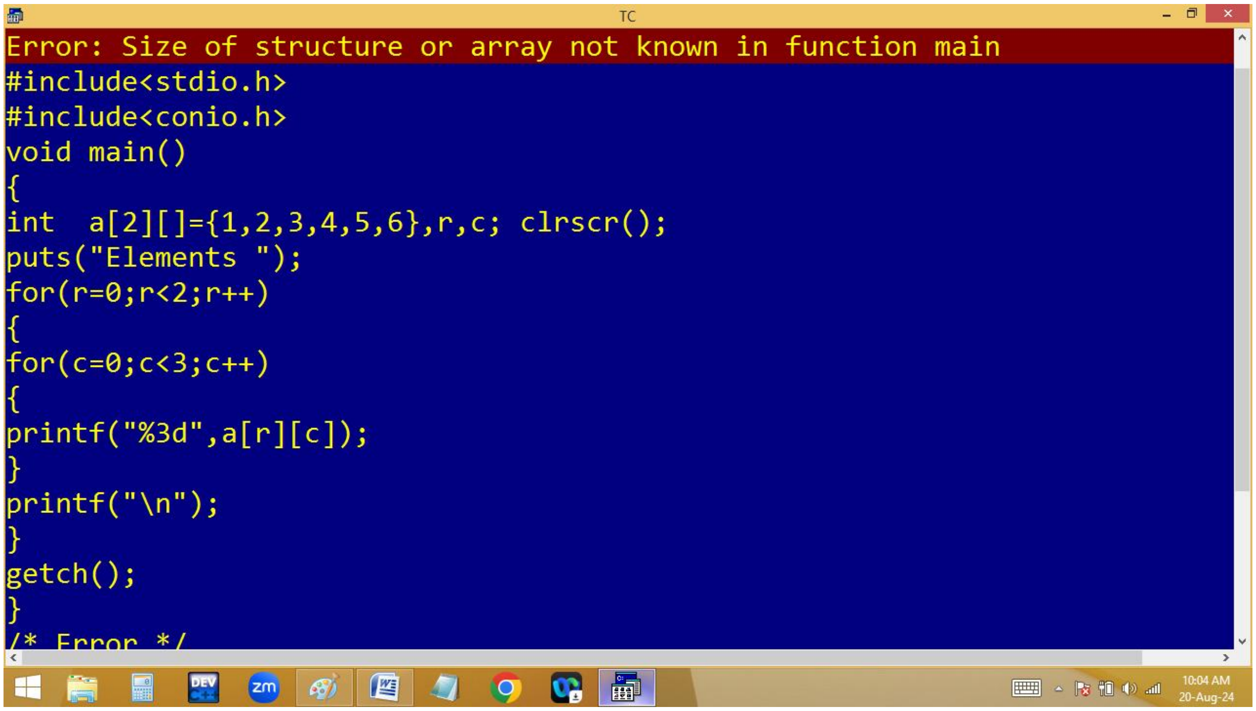
```
Elements
  1  2  0
  3  4  0
```



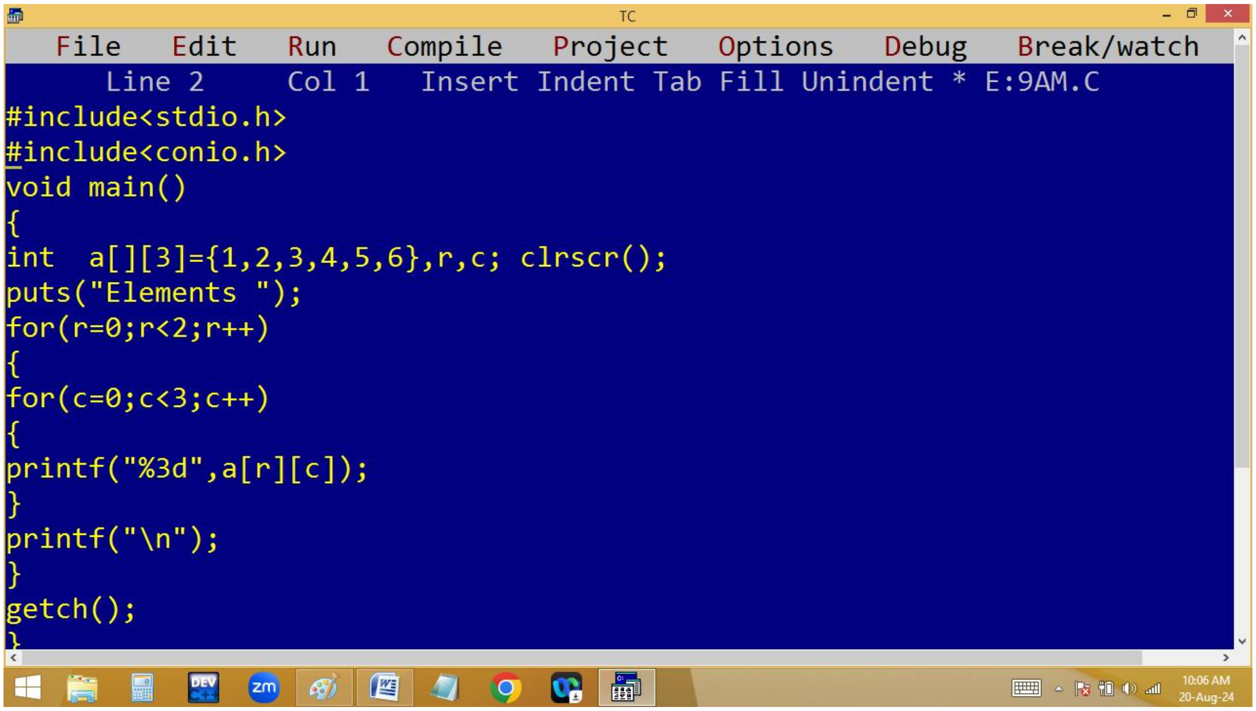
```
Error: Initializer syntax error in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={{1,2},{ }},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



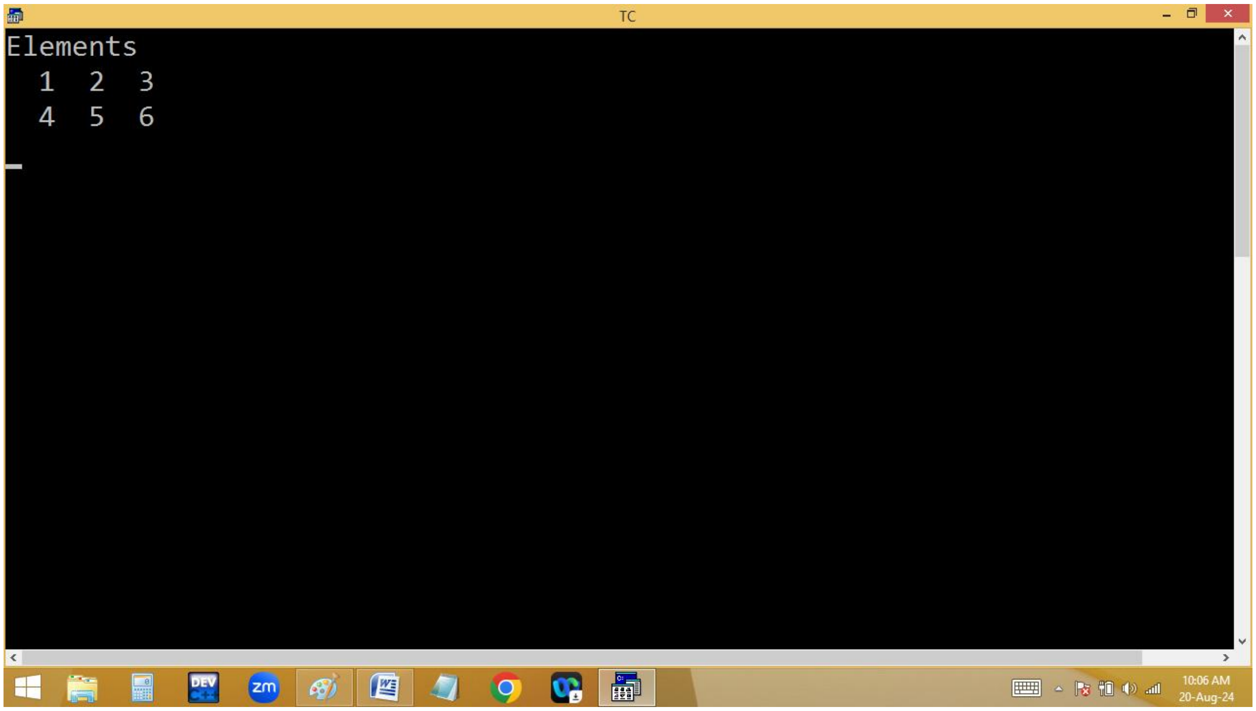
```
Error: Size of structure or array not known in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[][]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



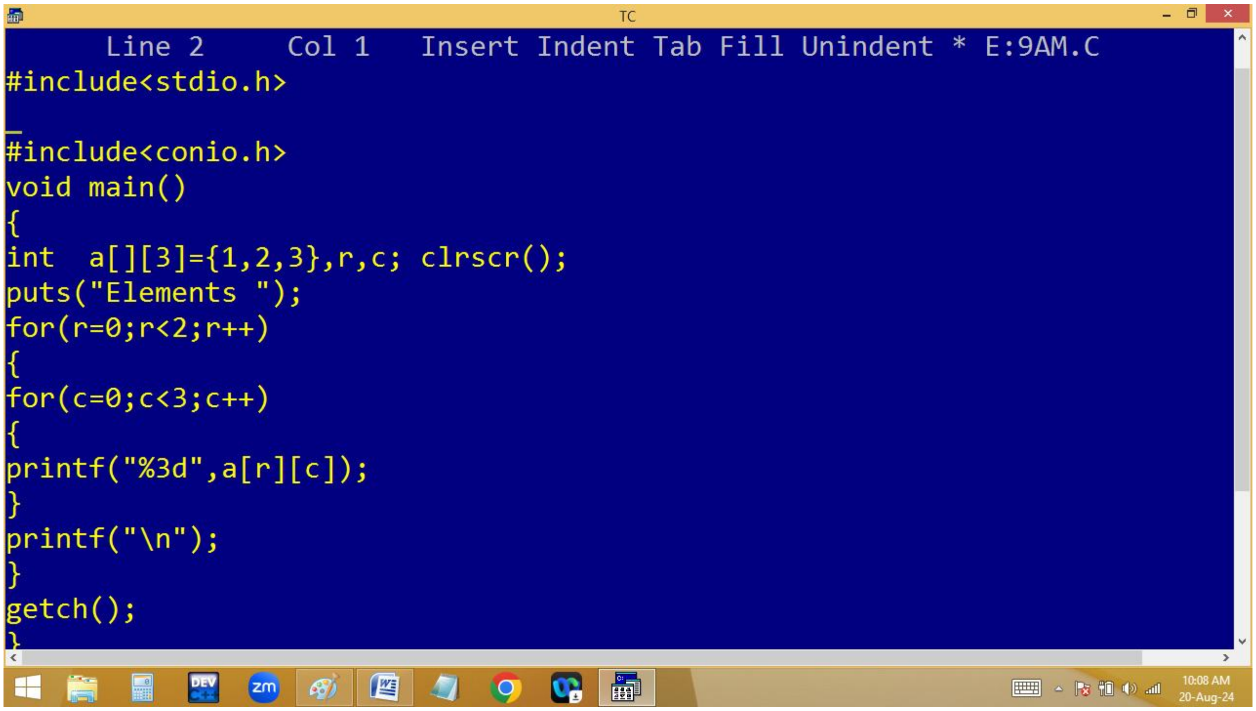
```
Error: Size of structure or array not known in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
/* Error */
```



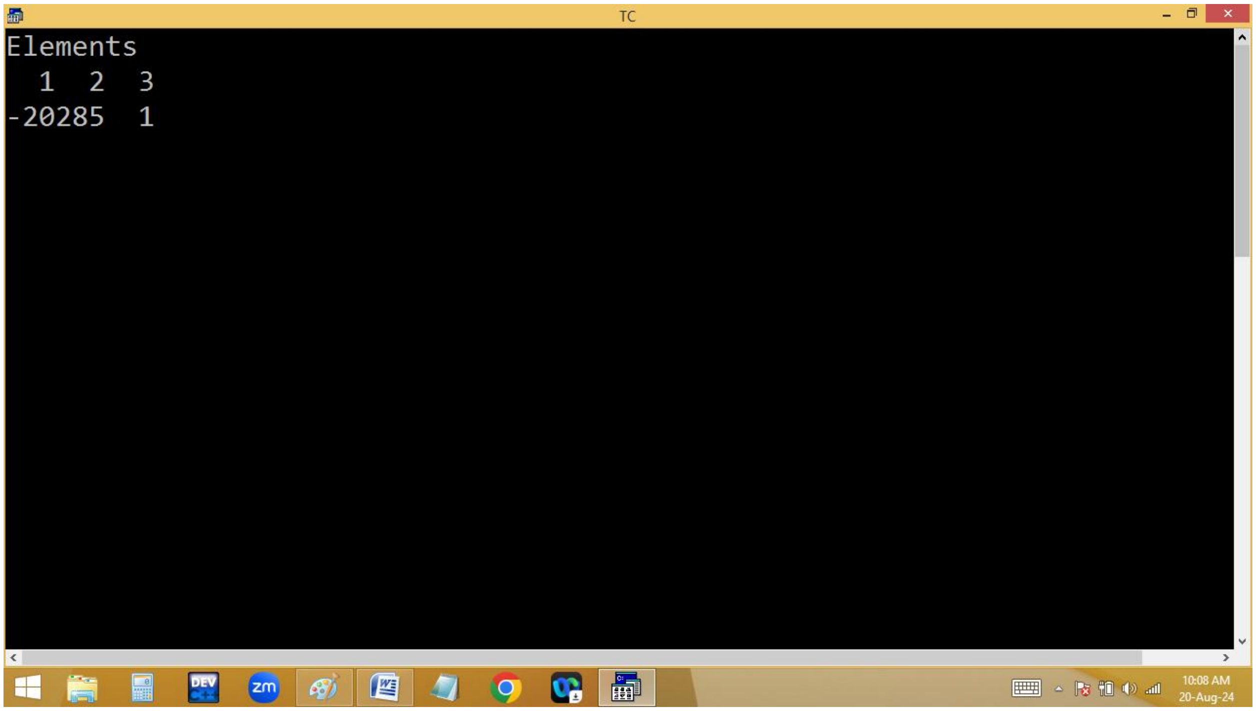
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[][3]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



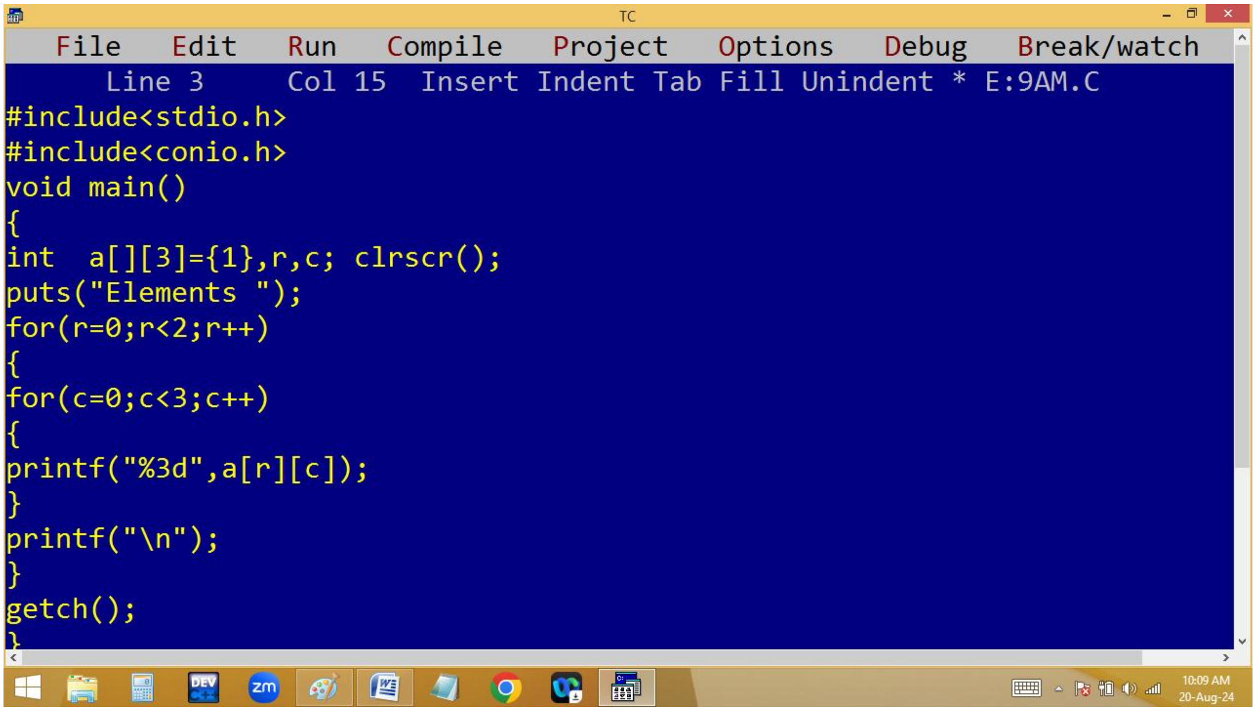
```
Elements
  1  2  3
  4  5  6
```



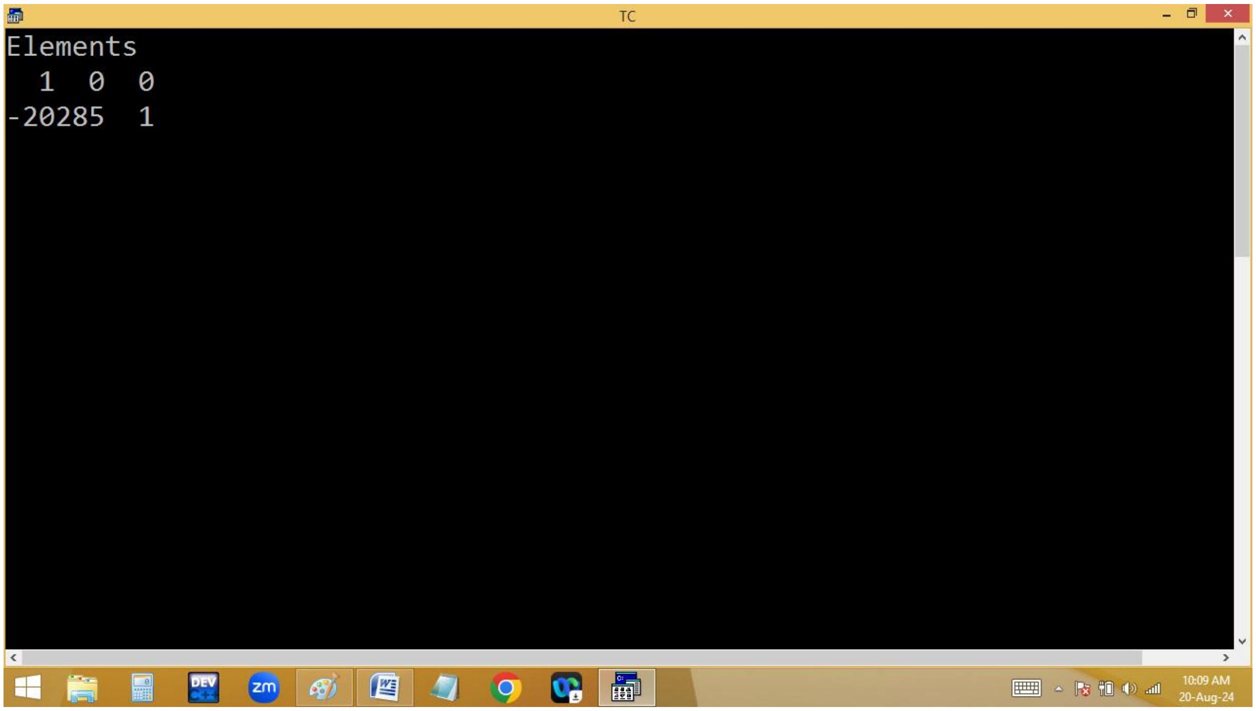
```
      Line 2      Col 1   Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>

#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();
}
```



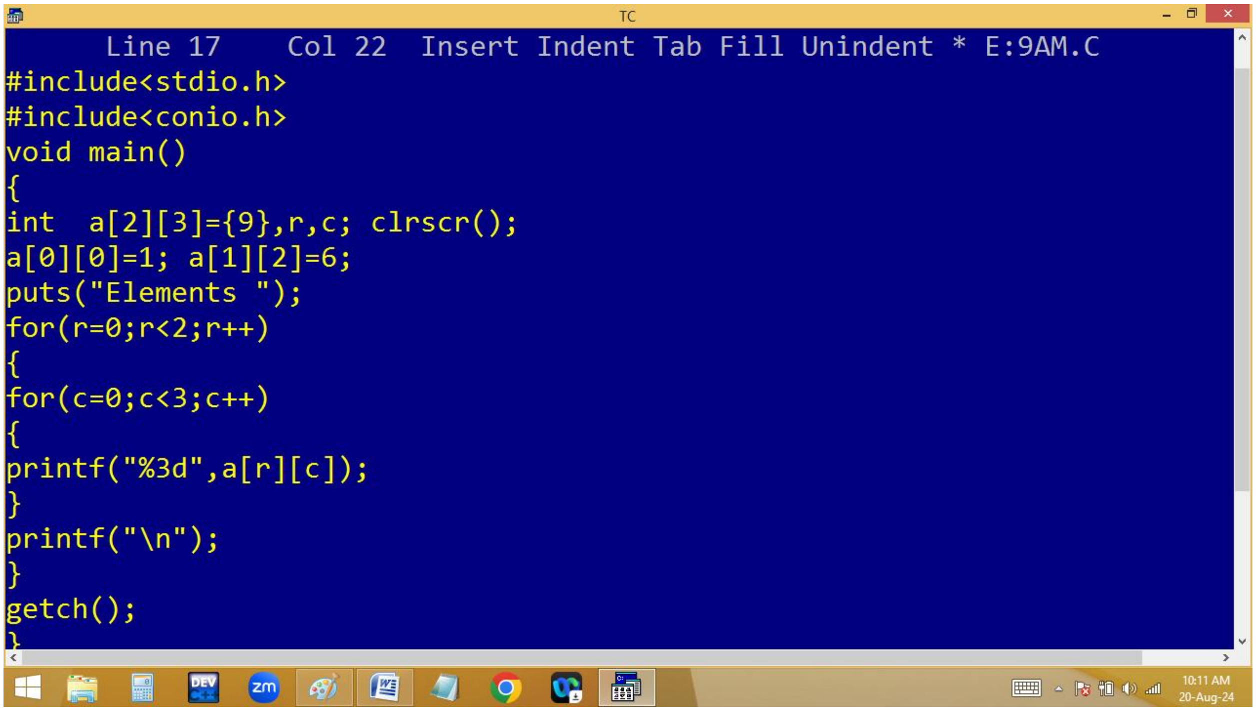
```
Elements
  1  2  3
-20285  1
```


TC
File Edit Run Compile Project Options Debug Break/watch
Line 3 Col 15 Insert Indent Tab Fill Unindent * E:9AM.C
10:09 AM
20-Aug-24

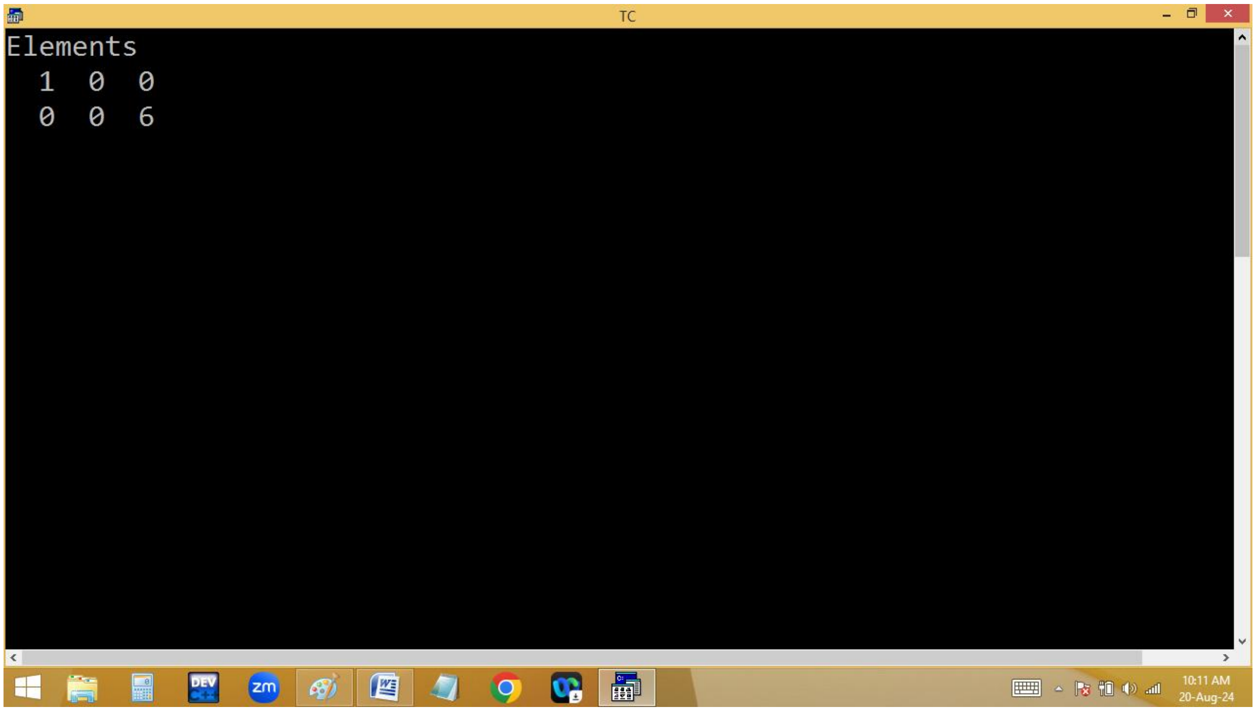
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[][3]={1},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



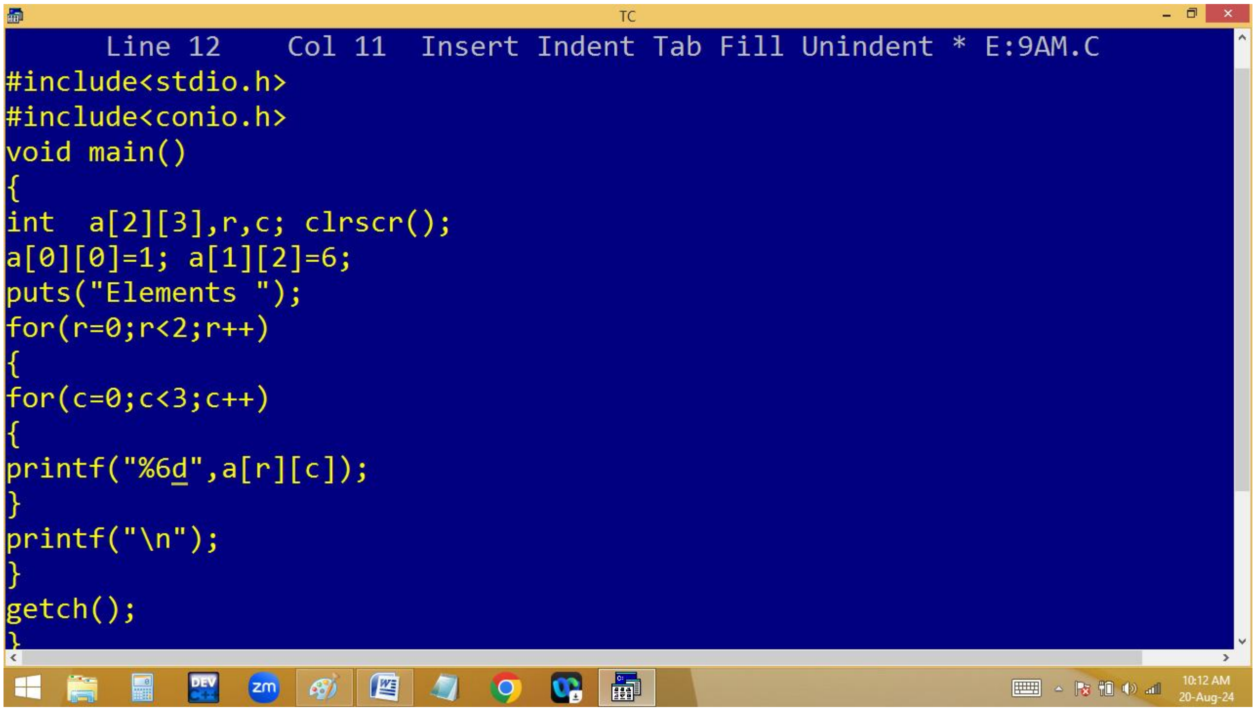
```
Elements
  1  0  0
-20285  1
```


TC
Line 17   Col 22 Insert Indent Tab Fill Unindent * E:9AM.C
10:11 AM
20-Aug-24

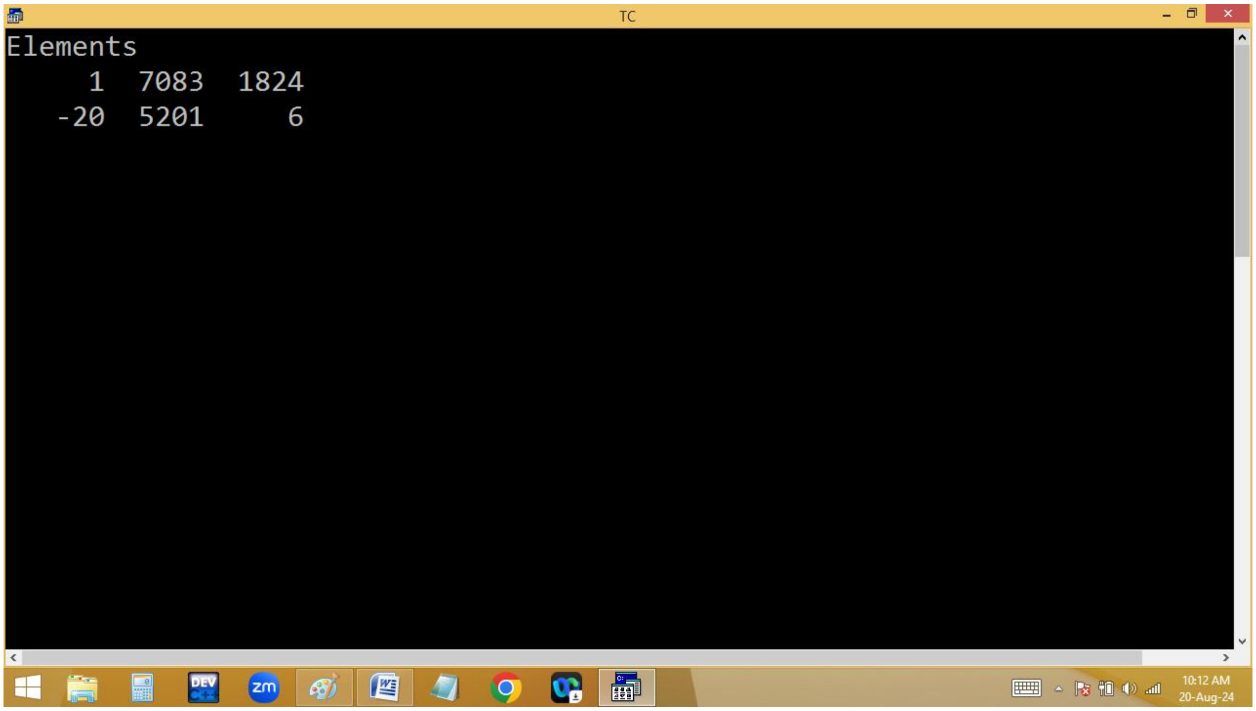
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3]={9},r,c; clrscr();
a[0][0]=1; a[1][2]=6;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%3d",a[r][c]);
}
printf("\n");
}
getch();
}
```



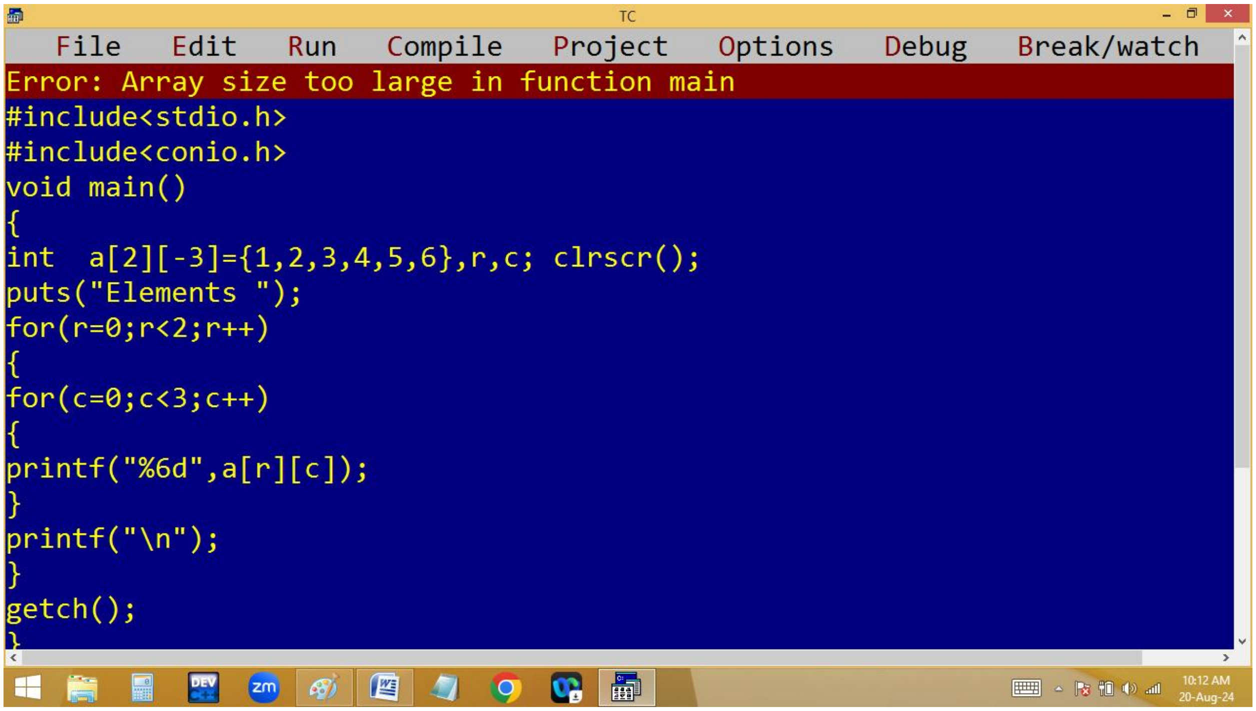
```
Elements
  1  0  0
  0  0  6
```



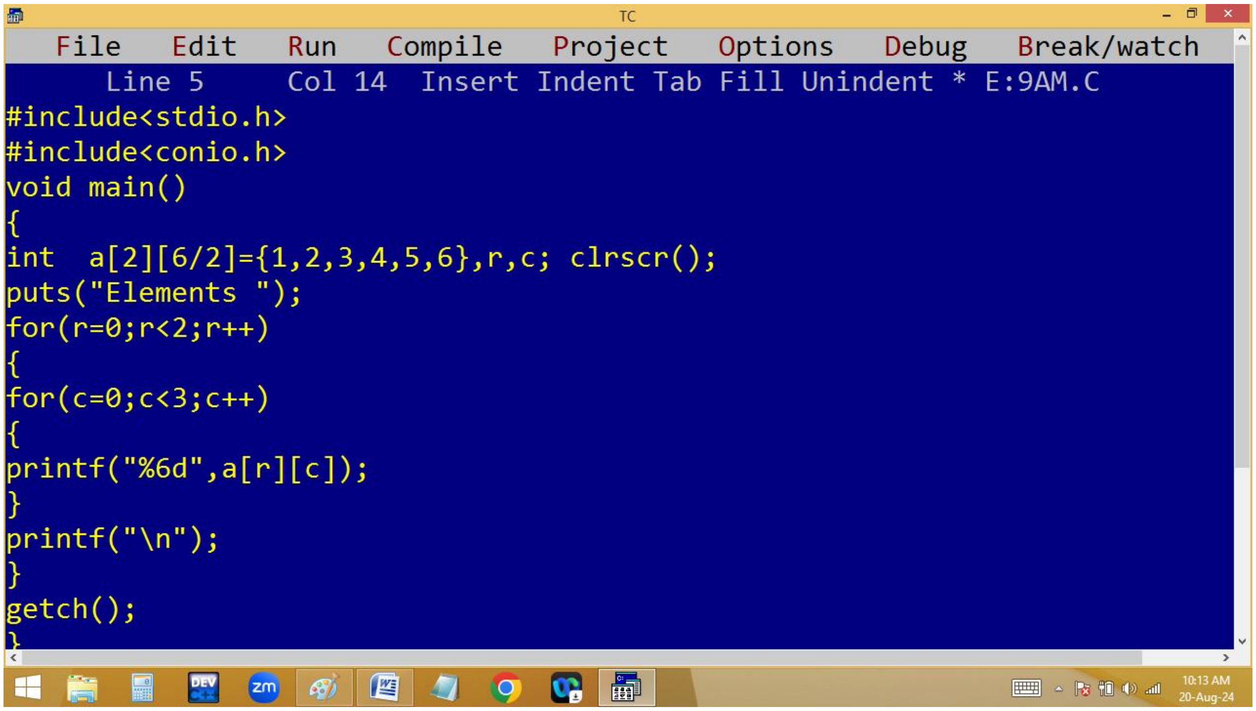
```
      Line 12     Col 11  Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][3],r,c; clrscr();
a[0][0]=1; a[1][2]=6;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```



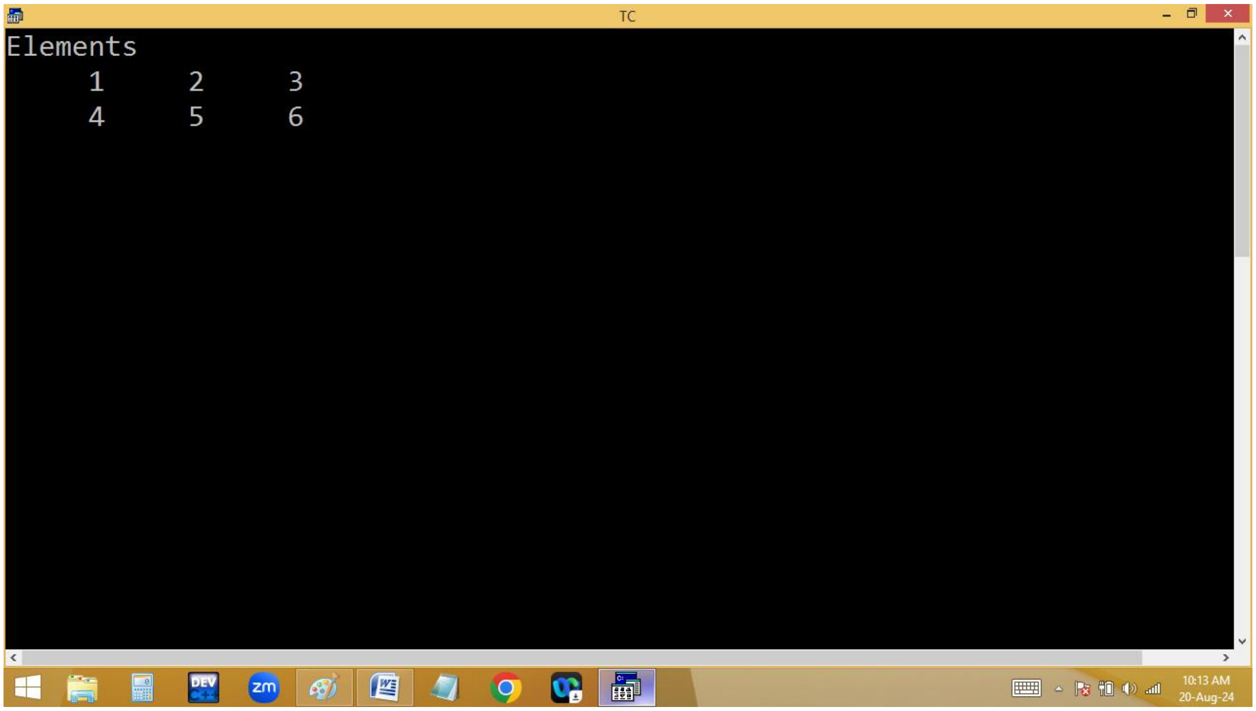
```
Elements
     1  7083  1824
   -20  5201     6
```



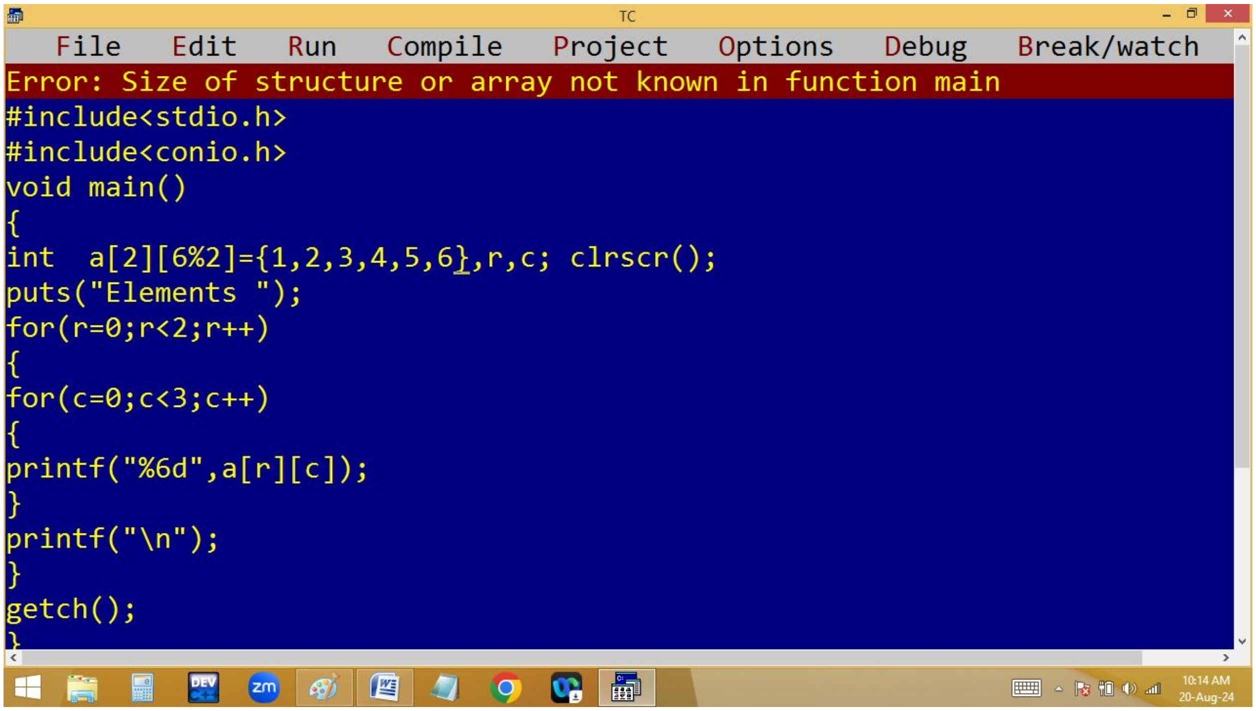
```
Error: Array size too large in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][-3]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```



```
     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();
}
```

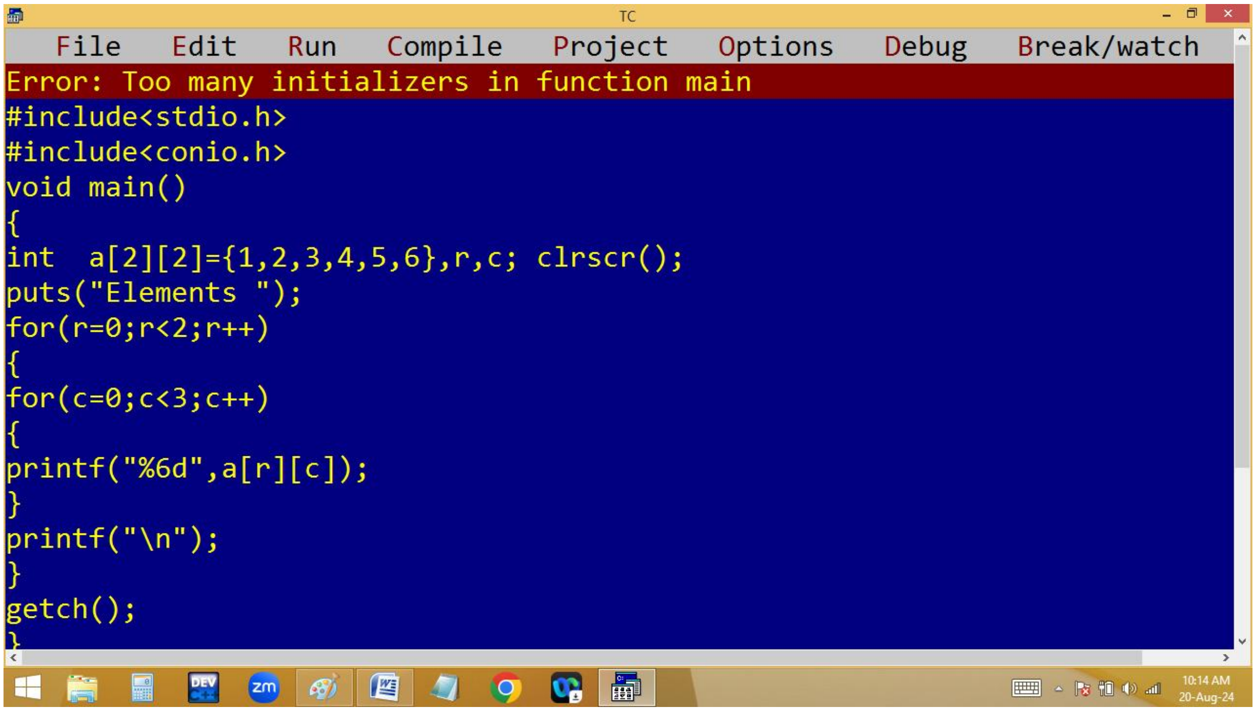


```
Elements
     1     2     3
     4     5     6
```

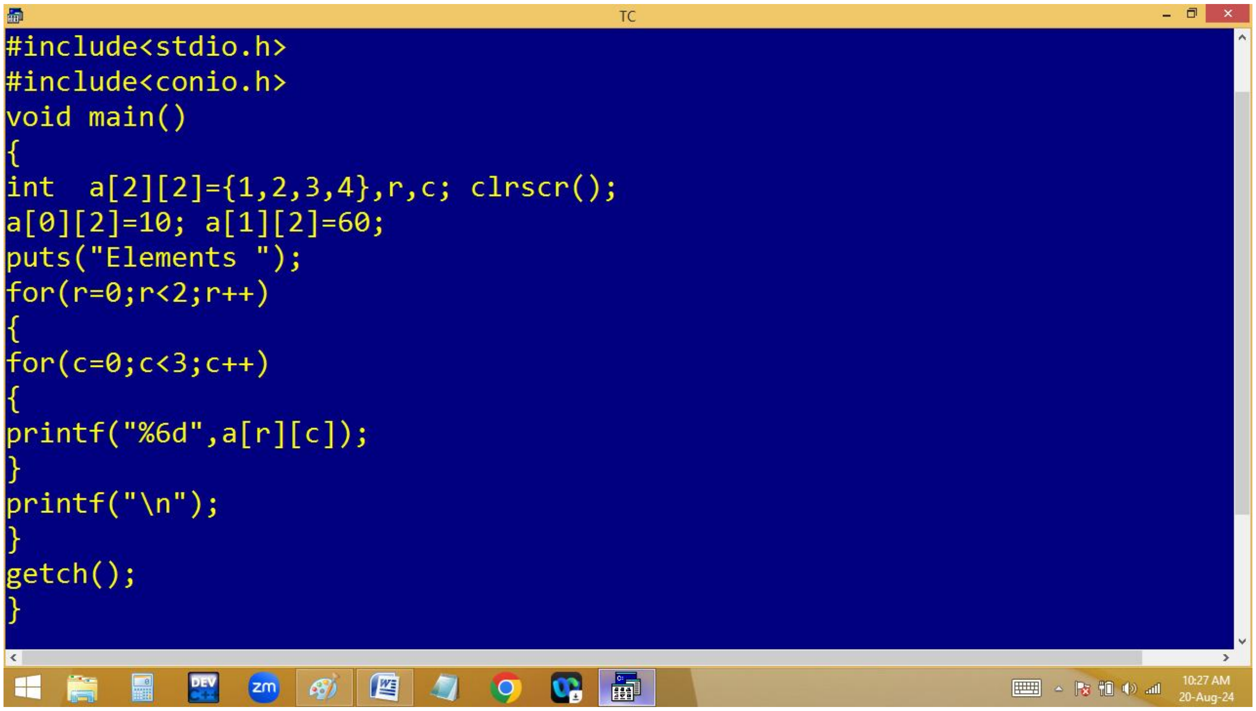


Error: Size of structure or array not known in function main

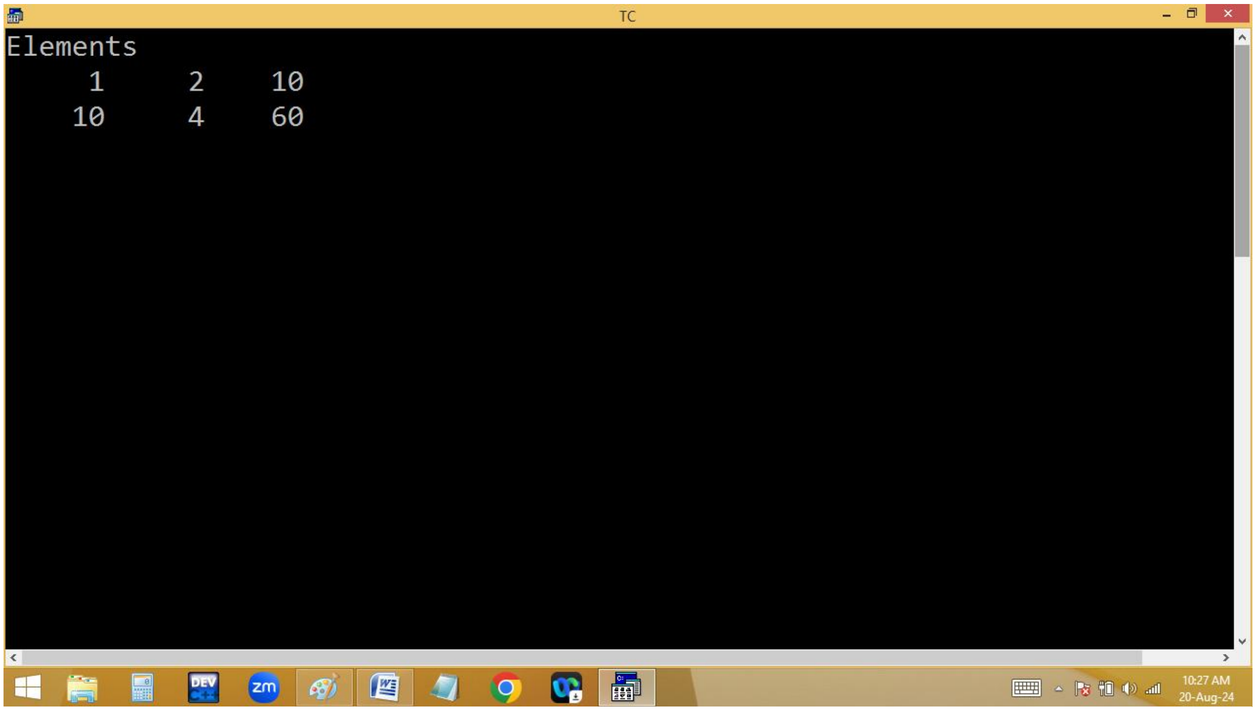
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][6%2]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```



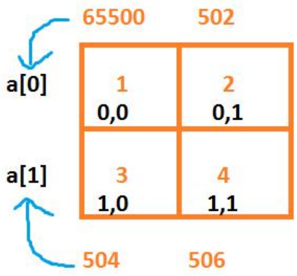
```
Error: Too many initializers in function main
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][2]={1,2,3,4,5,6},r,c; clrscr();
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

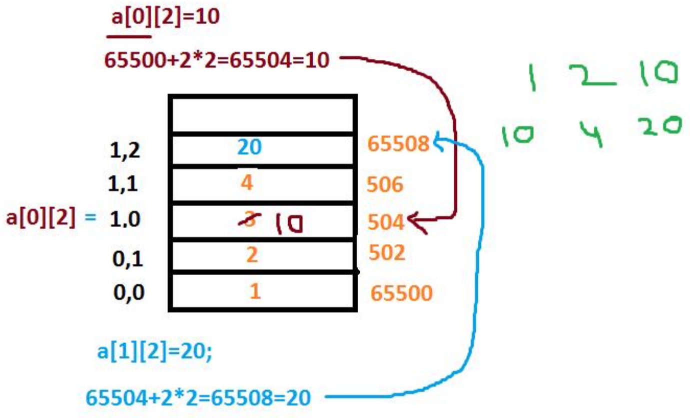


```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[2][2]={1,2,3,4},r,c; clrscr();
a[0][2]=10; a[1][2]=60;
puts("Elements ");
for(r=0;r<2;r++)
{
for(c=0;c<3;c++)
{
printf("%6d",a[r][c]);
}
printf("\n");
}
getch();
}
```

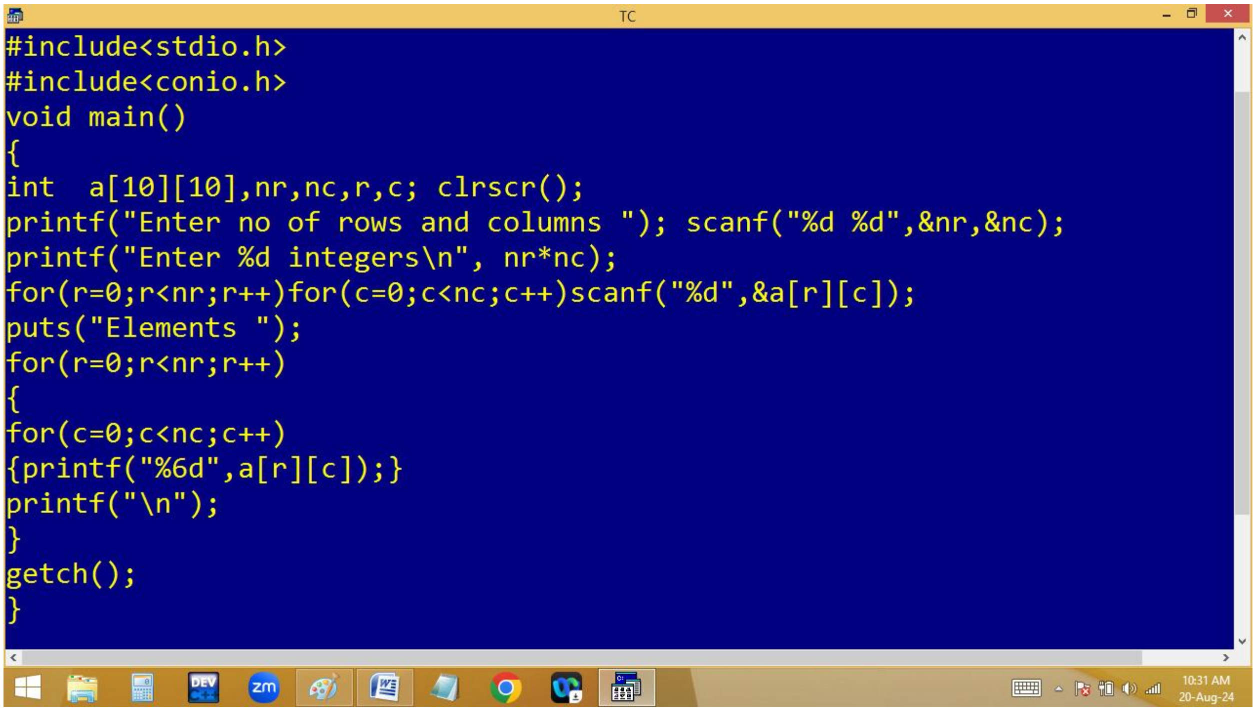


```
Elements
     1      2     10
    10      4     60
```

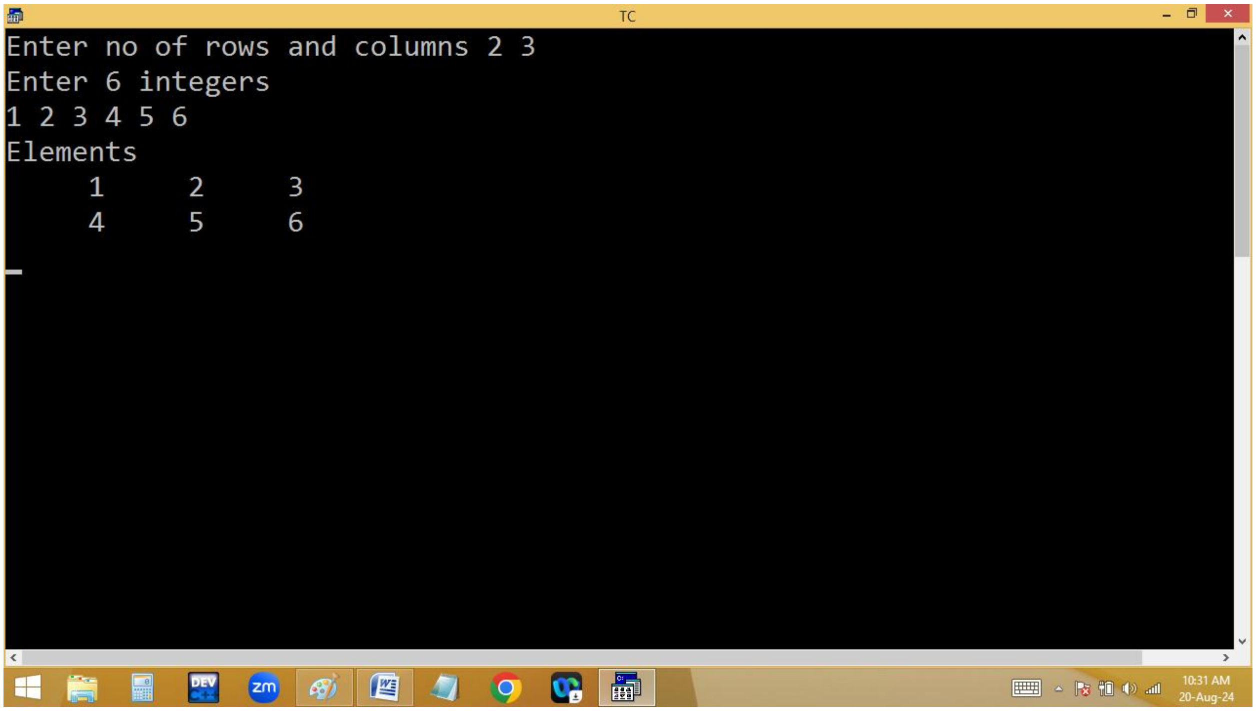




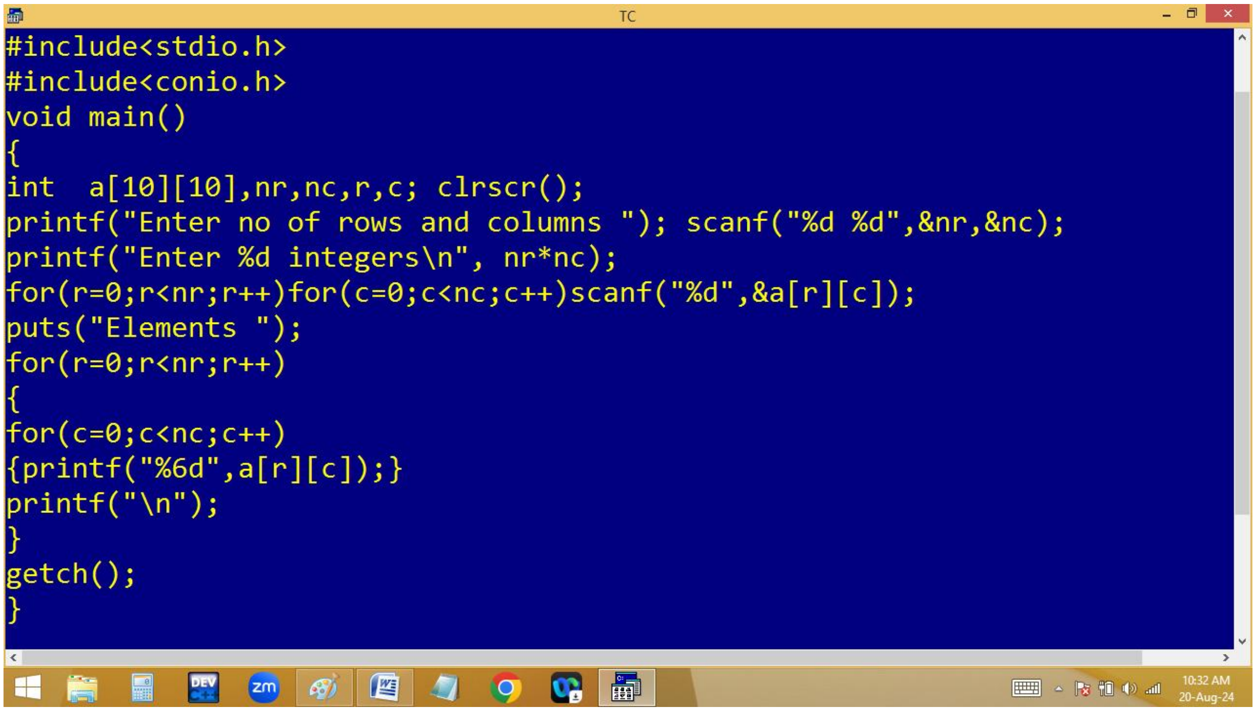
Reading and printing elements of a 2d array:



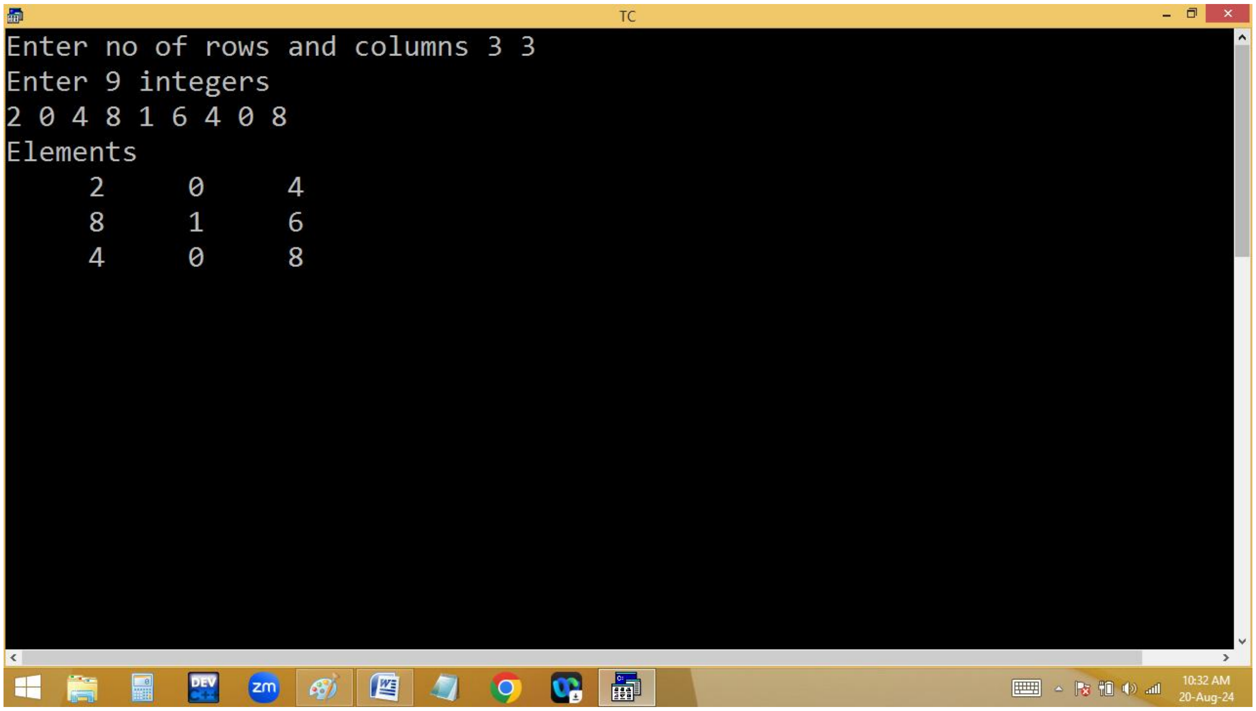
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("Elements ");
for(r=0;r<nr;r++)
{
for(c=0;c<nc;c++)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



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

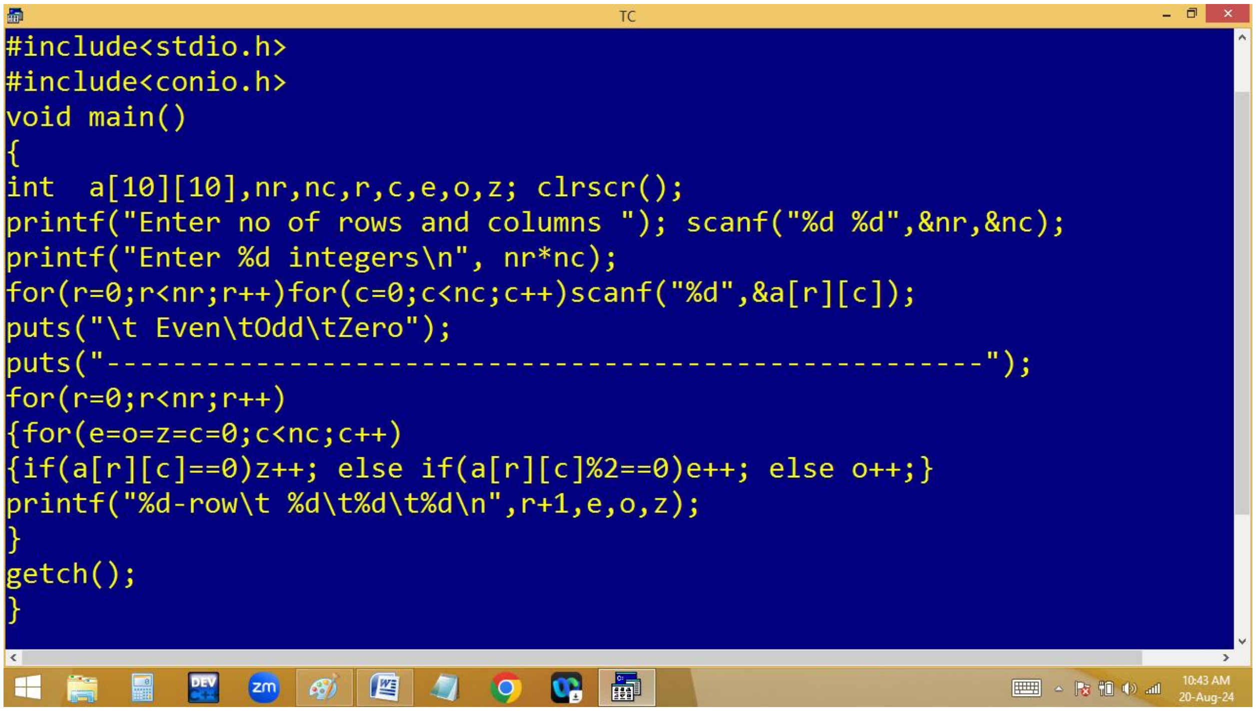


```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("Elements ");
for(r=0;r<nr;r++)
{
for(c=0;c<nc;c++)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

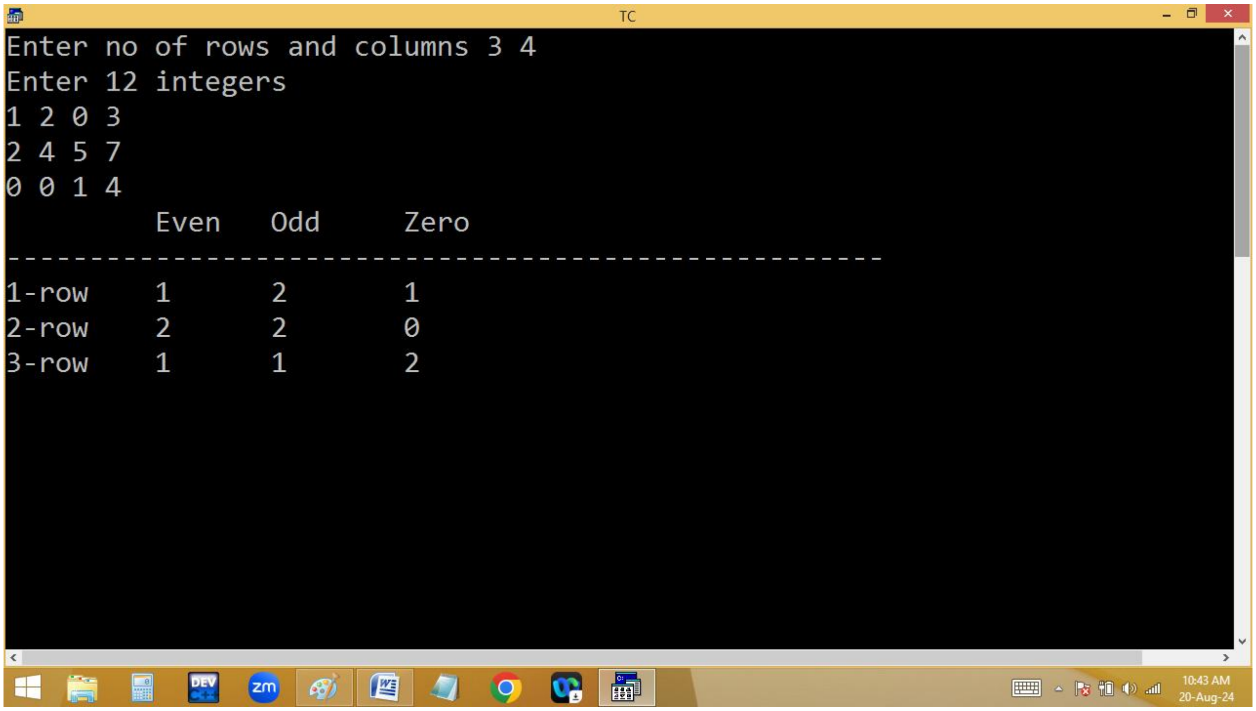


```
Enter no of rows and columns 3 3
Enter 9 integers
2 0 4 8 1 6 4 0 8
Elements
     2     0     4
     8     1     6
     4     0     8
```

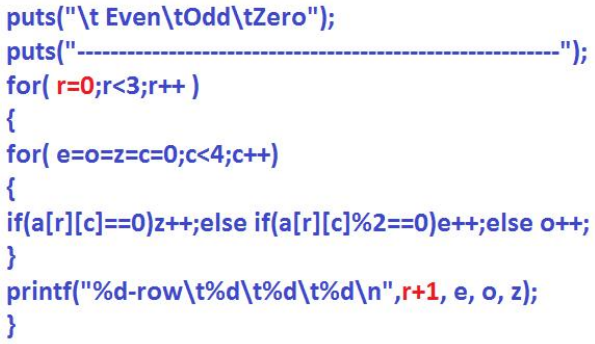
Read n elements to n*n array and find the no of even, odd, zero elements row wise.



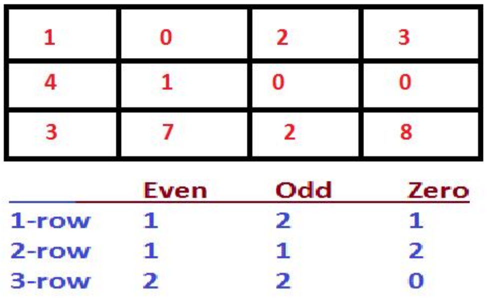
```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a[10][10],nr,nc,r,c,e,o,z; clrscr();
printf("Enter no of rows and columns "); scanf("%d %d",&nr,&nc);
printf("Enter %d integers\n", nr*nc);
for(r=0;r<nr;r++)for(c=0;c<nc;c++)scanf("%d",&a[r][c]);
puts("\t Even\tOdd\tZero");
puts("-----------------------------------------------------");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