

ARRAYS

It is collection of homogeneous [same type] variables.

Array is nothing but collection of contiguous memory locations, where we can store and manage more than one value of same type under one name.

It is a derived data type.

It is an implicit / internal pointer.

It is implicit const pointer

It is one of the data structure.

Advantages:

Generally to store several values of same type, we have to declare several variables. Here we have to remember all these variable names also. When the program is too big, it is very difficult to remember all the variable names. In this situation, the only solution is array.

Array reduce program length.

Array minimize the errors.

In functions to carry several values of same type at a time, we are using arrays.

Disadvantage: Array size is static. i.e. array size should be entered at design time and it is fixed at compile time. Due to this we can't change this array size at runtime, which causes memory wastage some times.

In C language we are using

1. One dimensional arrays
2. Multi dimensional arrays

One dimensional arrays:

- An array with one row and several columns.
- An array with single subscripting operator **[]** is called one dimensional array.
- It is an implicit single pointer.

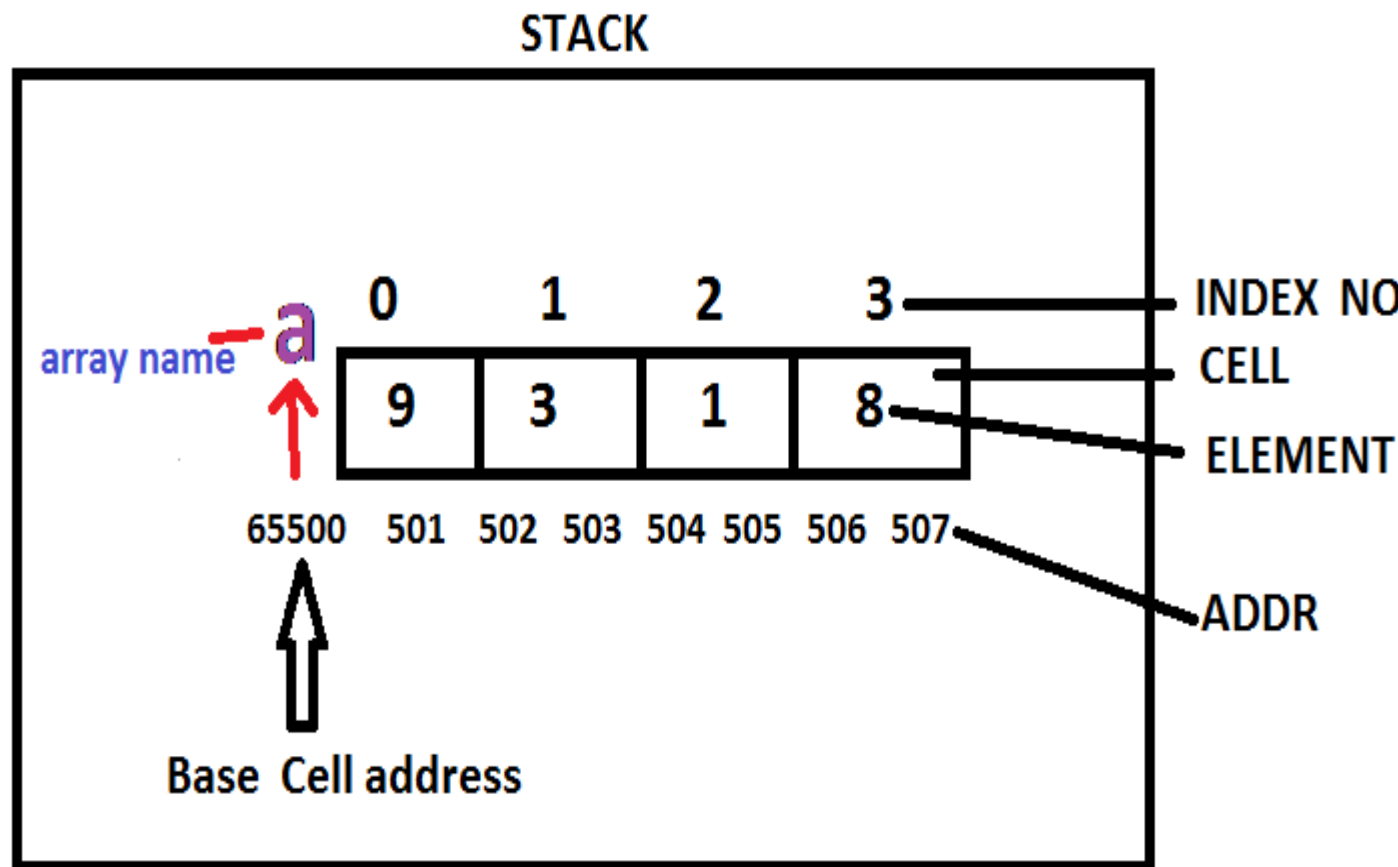
Syntax:

datatype variable[size] = {elements};

Eg:

int a[4] = { 9, 3, 1, 8 };

Memory allocation for array:



Array is implicit pointer because of array variable stores base cell [0 cell 1st byte] address. Hence array variable value and 0 cell address both are same.

Array declaration methods:

`int a[3];` Ok

`int a[];` No

`int a[3]={1,2,3};` Ok

`int a[]={1,2,3};` Ok

`int a[0]={1,2,3};` Ok

`int a[-5];` No

`int a[5.5];` No

`int n = 5, a[n];` No

`int a[3]={10,20};` Ok

`int a[3]={1, 2, 3, 4};` No

`int a[0];` error

`#define n 5 /* macro */`

`int a[n];` Ok

`const int n=5, a[n];` No

`int a[5>3];` → `int a[1];` Ok

`int a[3<2];` → `int a[0];` No

`int a[2+3];` → `int a[5];` Ok

`int a[5%3];` → `int a[2];` Ok

`int a[5%5];` → `int a[0];` No

`int a[1,2,3];` → error

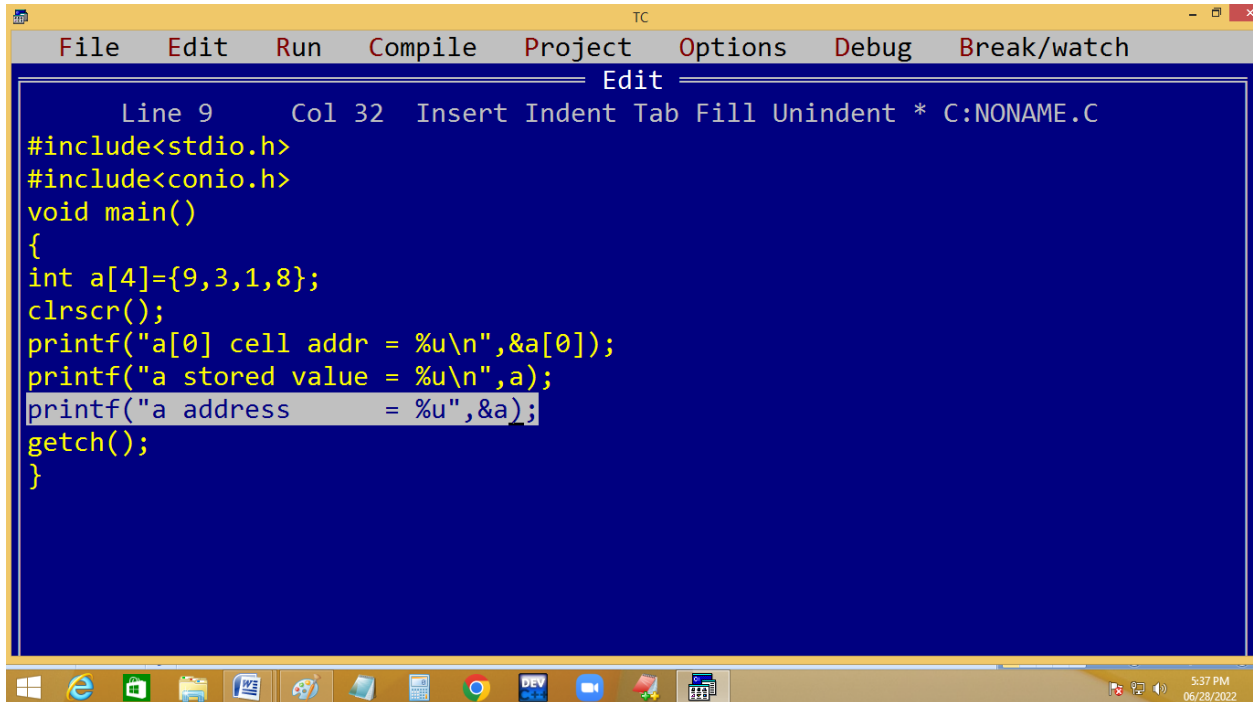
`int a[40000];` → $40000 * 2 = 80000$ bytes → No

Note: Stack size is 65536 bytes(64kb) Only.

`float a[10000];` Ok → $10000 * 4 = 40000$ bytes

`float a[20000];` No → $20000 * 4 = 80000$ bytes

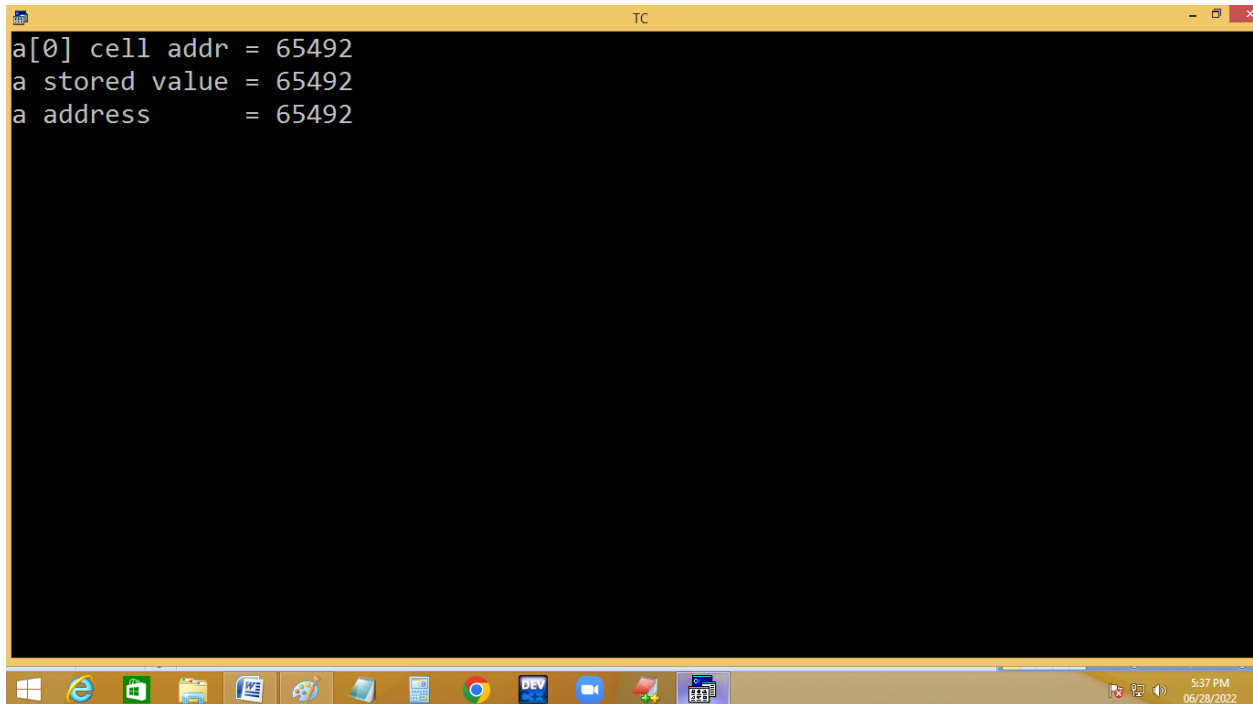
Eg. finding array address.



The screenshot shows the Turbo C++ (TC) editor window. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 9 Col 32 Insert Indent Tab Fill Unindent * C:NONAME.C'. The code in the editor is as follows:

```
Line 9 Col 32 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
    int a[4]={9,3,1,8};
    clrscr();
    printf("a[0] cell addr = %u\n",&a[0]);
    printf("a stored value = %u\n",a);
    printf("a address      = %u",&a);
    getch();
}
```

The taskbar at the bottom shows various application icons and the system clock indicating 5:37 PM on 06/28/2022.

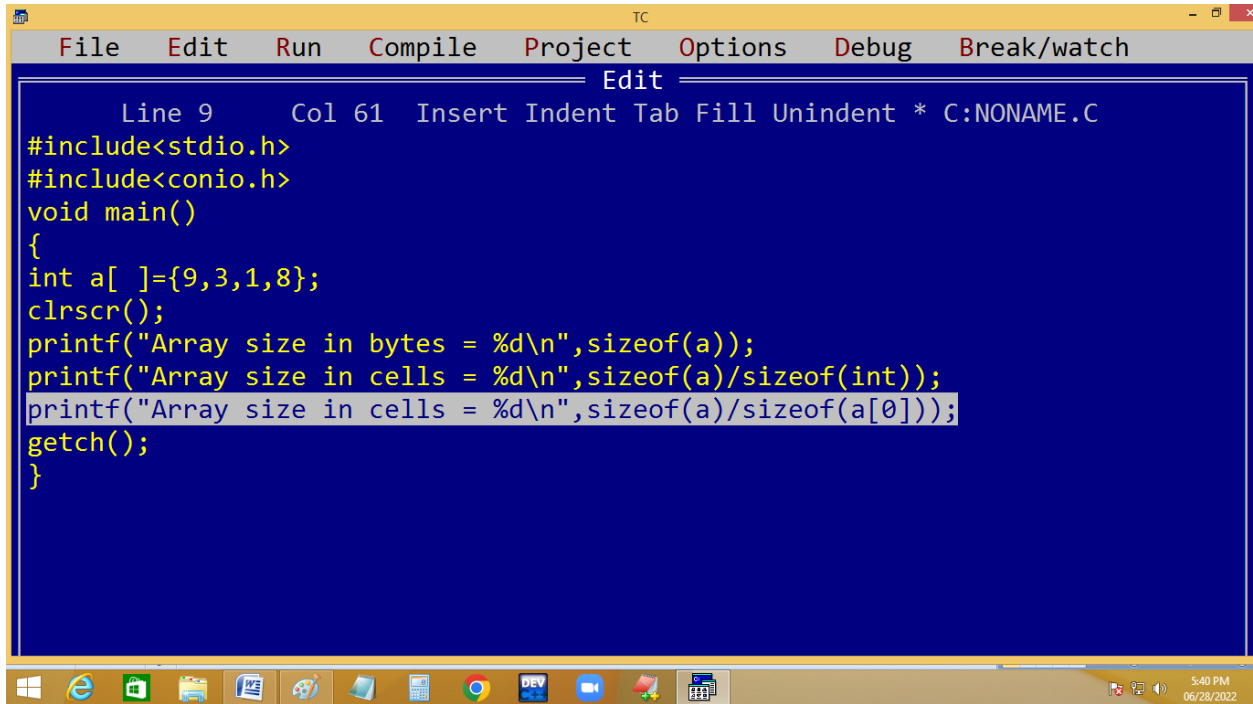


The screenshot shows the Turbo C++ (TC) editor window after execution. The output displayed is:

```
a[0] cell addr = 65492
a stored value = 65492
a address      = 65492
```

The taskbar at the bottom shows various application icons and the system clock indicating 5:37 PM on 06/28/2022.

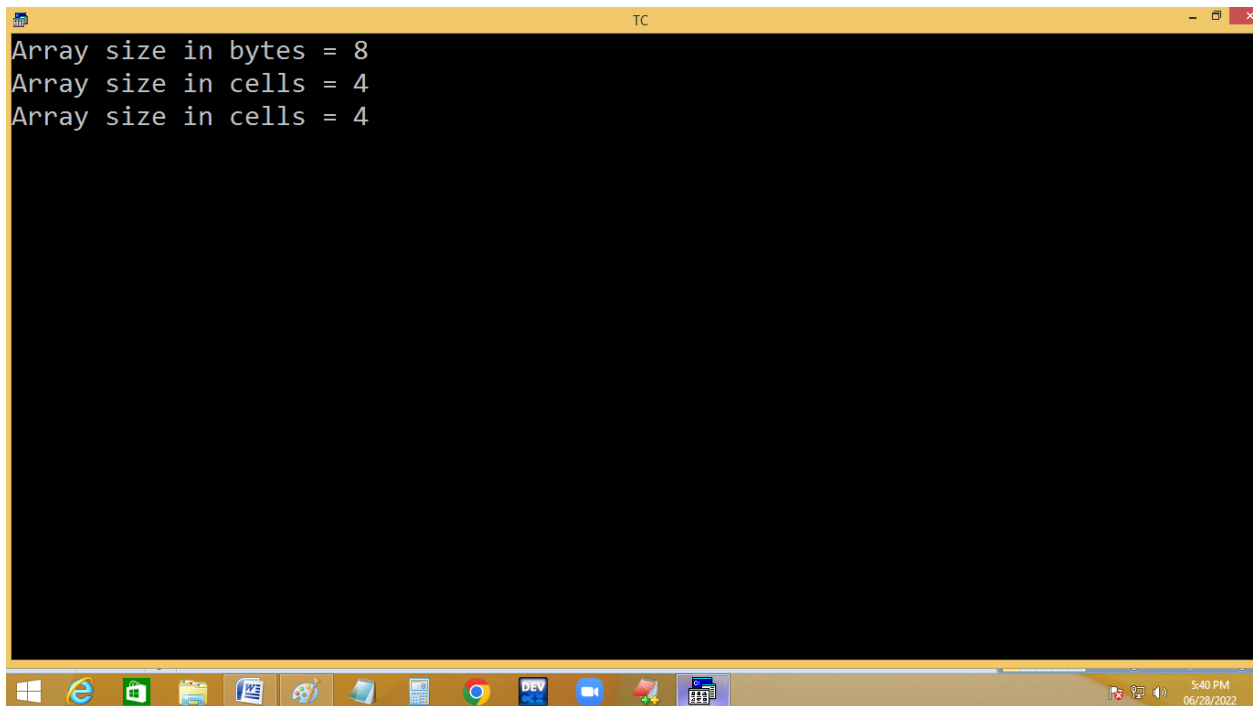
Eg. Finding array size:



The screenshot shows the Turbo C++ (TC) editor window. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 9', 'Col 61', and the file path 'C:\NONAME.C'. The code in the editor is as follows:

```
Line 9      Col 61  Insert Indent Tab Fill Unindent * C:\NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[ ]={9,3,1,8};
clrscr();
printf("Array size in bytes = %d\n",sizeof(a));
printf("Array size in cells = %d\n",sizeof(a)/sizeof(int));
printf("Array size in cells = %d\n",sizeof(a)/sizeof(a[0]));
getch();
}
```

The taskbar at the bottom shows various application icons and the system clock indicating 5:40 PM on 06/28/2022.

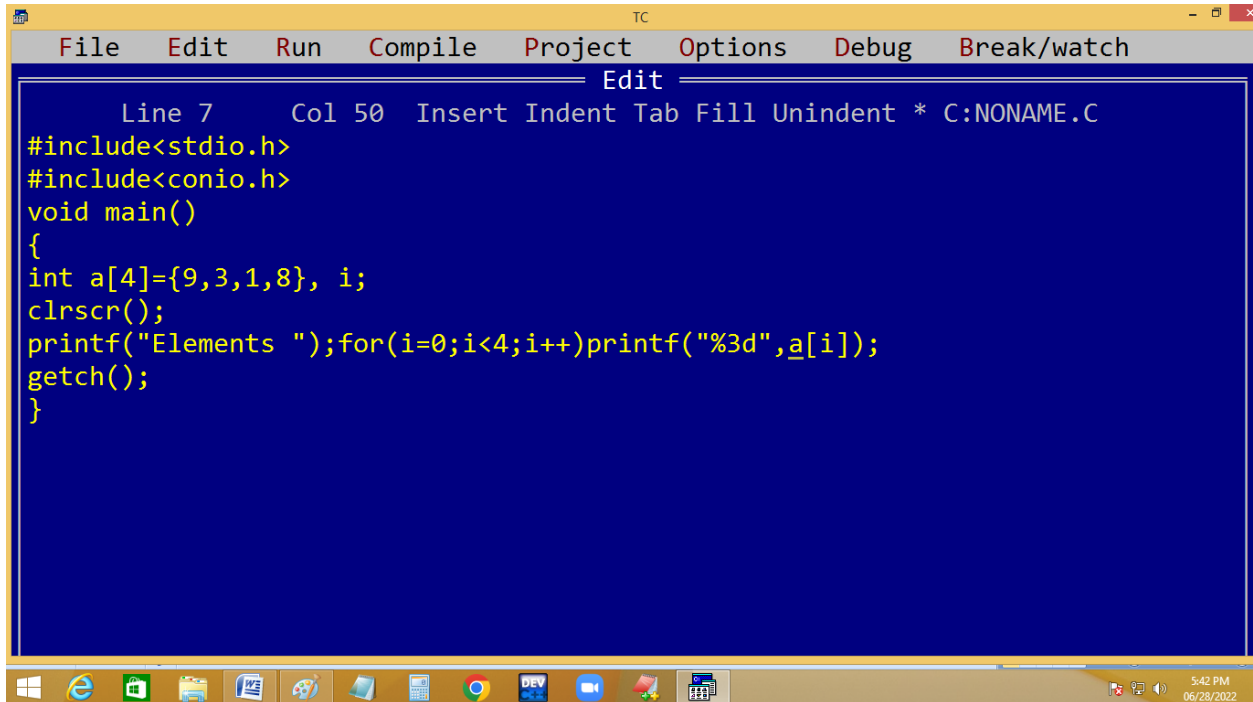


The screenshot shows the Turbo C++ (TC) console window. The output of the program is displayed as follows:

```
Array size in bytes = 8
Array size in cells = 4
Array size in cells = 4
```

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

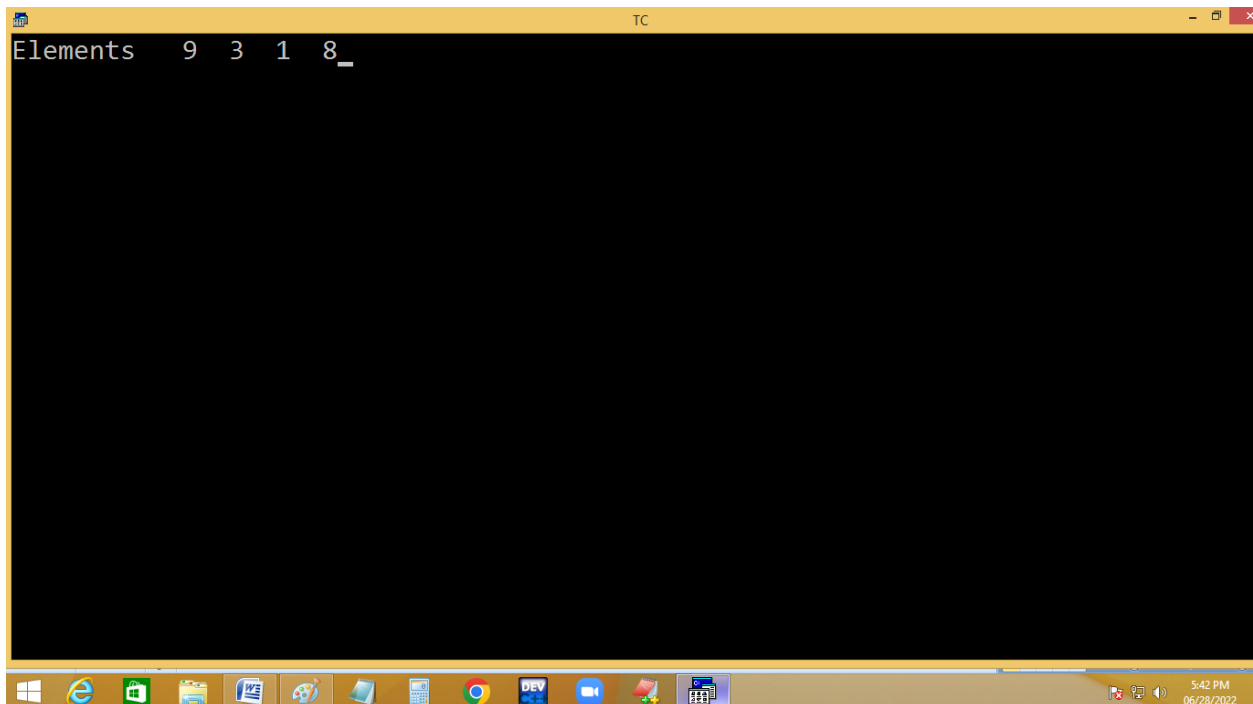
Direct initialization of array elements:



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

```
Line 7      Col 50  Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
  int a[4]={9,3,1,8}, i;
  clrscr();
  printf("Elements ");for(i=0;i<4;i++)printf("%3d",a[i]);
  getch();
}
```

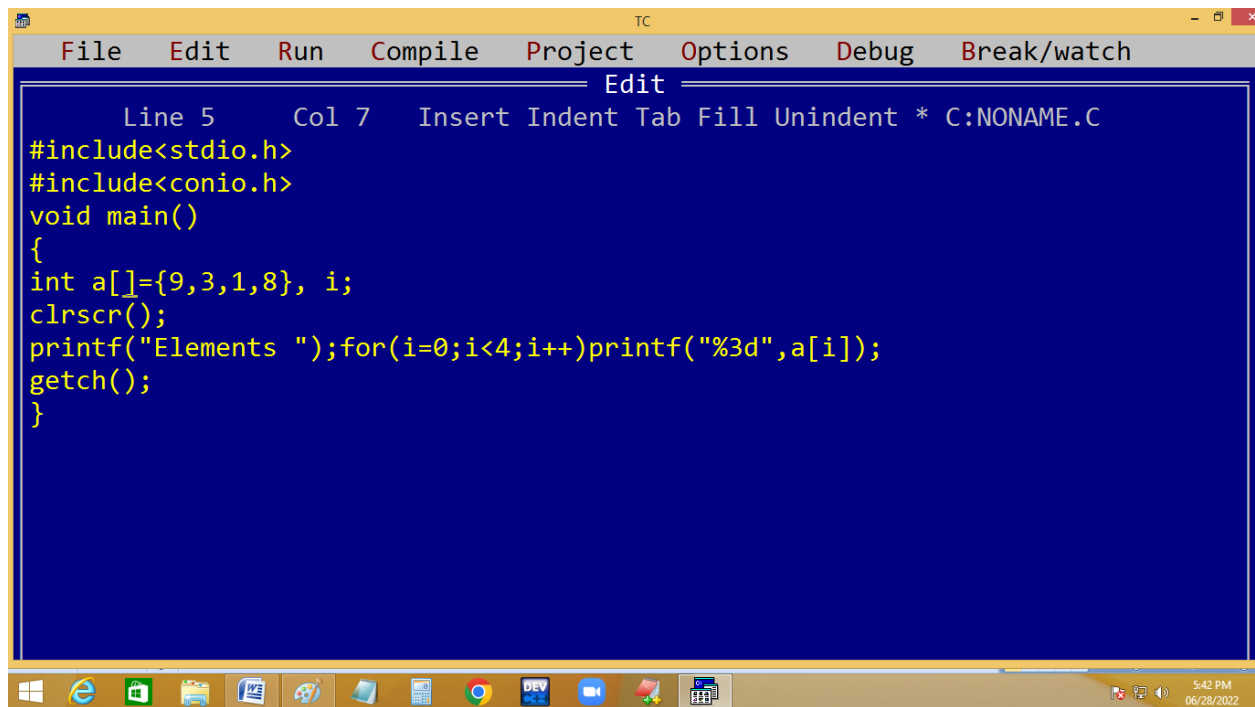
The IDE's menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the bottom indicates the time as 5:42 PM on 06/28/2022.



The screenshot shows the Turbo C++ (TC) IDE with the output of the program in the console window:

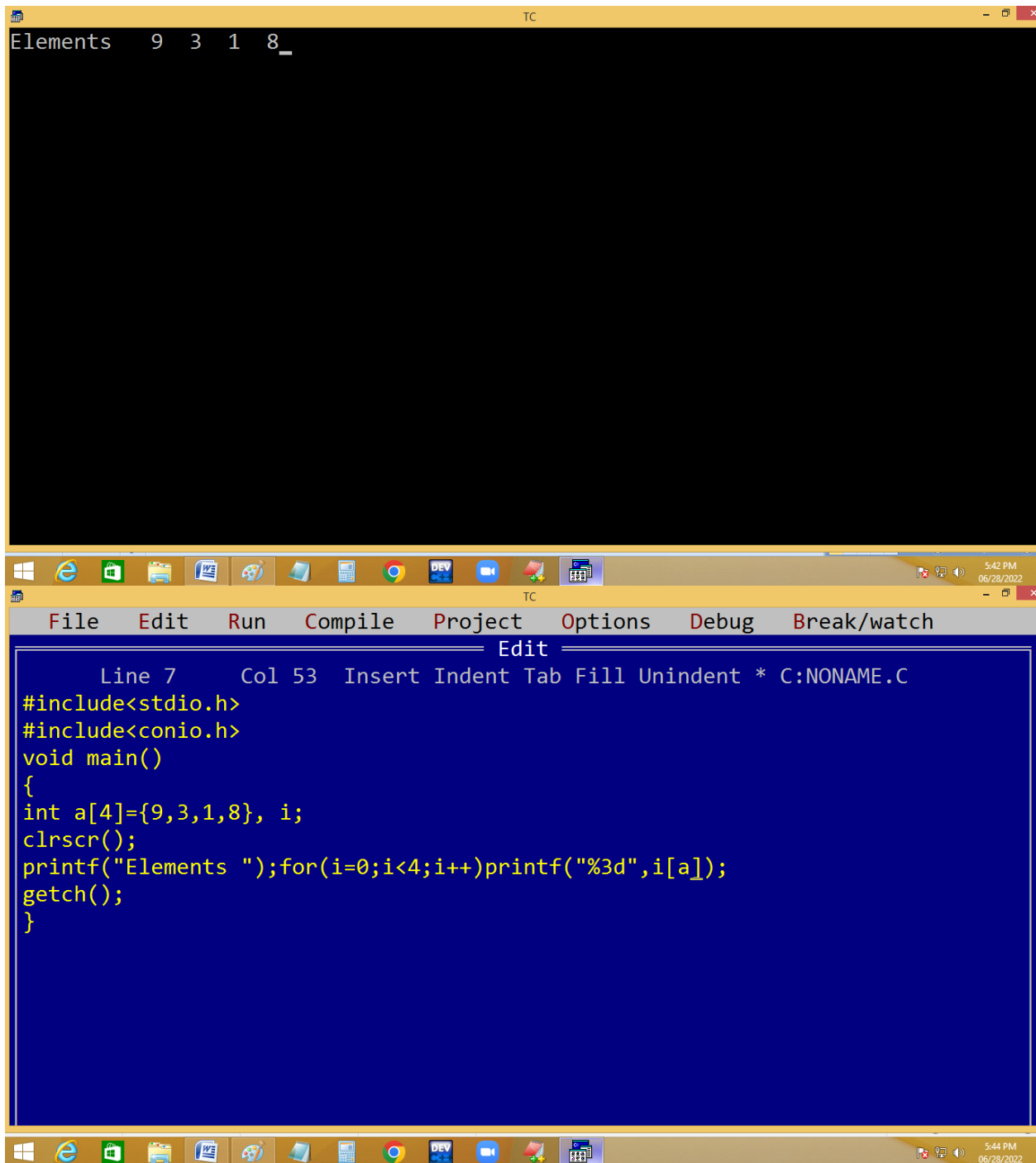
```
Elements  9  3  1  8_
```

The IDE's menu bar and status bar are the same as in the previous screenshot, showing the time as 5:42 PM on 06/28/2022.



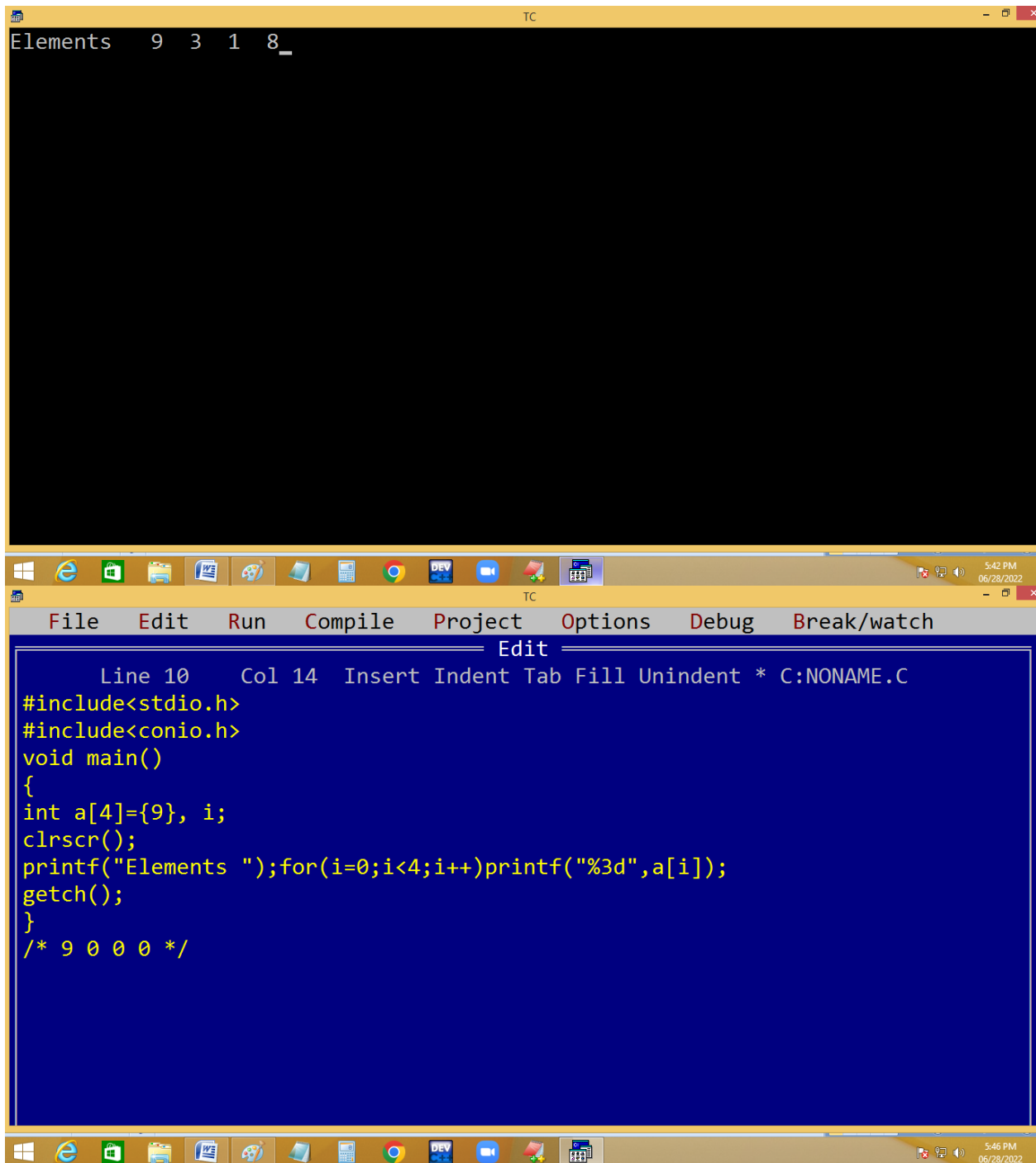
The image shows a screenshot of a Turbo C++ (TC) IDE window. The window has a yellow title bar with the text "TC" and standard window controls. Below the title bar is a menu bar with the following options: File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The main editing area has a dark blue background with yellow text. At the top of the editing area, there is a status bar that reads "Line 5 Col 7 Insert Indent Tab Fill Unindent * C:NONAME.C". The code being edited is a C program that includes `<stdio.h>` and `<conio.h>`, defines a `main()` function, declares an array `a` with values {9, 3, 1, 8}, clears the screen with `clrscr()`, and prints the elements of the array using a `for` loop and `printf`. The code ends with `getch()` and a closing brace for the `main` function. The Windows taskbar is visible at the bottom of the screen, showing various application icons and the system clock indicating 5:42 PM on 06/28/2022.

```
Line 5 Col 7 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[]={9,3,1,8}, i;
clrscr();
printf("Elements ");for(i=0;i<4;i++)printf("%3d",a[i]);
getch();
}
```

The image shows a screenshot of a Turbo C++ (TC) IDE. The top window, titled 'TC', displays the output of a program: 'Elements 9 3 1 8_'. The bottom window, titled 'Edit', shows the source code of the program. The code includes headers for `stdio.h` and `conio.h`, and defines a `main` function. Inside `main`, an array `a` of size 4 is initialized with the values {9, 3, 1, 8}. The program then clears the screen with `clrscr()`, prints the label 'Elements ' followed by a space, and uses a `for` loop to print each element of the array in a 3-digit format. The cursor is positioned at the end of the output line in the top window.

```
Line 7      Col 53  Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[4]={9,3,1,8}, i;
clrscr();
printf("Elements ");for(i=0;i<4;i++)printf("%3d",i[a]);
getch();
}
```



The image shows a screenshot of the Turbo C++ (TC) IDE. The main window displays the output of a program: "Elements 9 3 1 8_". The status bar at the bottom of the window indicates "Line 10 Col 14 Insert Indent Tab Fill Unindent * C:\NONAME.C". The code editor shows the following C program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a[4]={9}, i;
    clrscr();
    printf("Elements ");for(i=0;i<4;i++)printf("%3d",a[i]);
    getch();
}
/* 9 0 0 0 */
```

The Windows taskbar at the bottom shows the time as 5:42 PM on 06/28/2022. The system tray includes icons for network, volume, and power.

TC

File Edit Run Compile Project Options Debug Break/watch

Edit

Line 11 Col 14 Insert Indent Tab Fill Unindent * C:NONAME.C

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a[4], i;
a[0]=9;
clrscr();
printf("Elements ");for(i=0;i<4;i++)printf("%6d",a[i]);
getch();
}
/* 9 gr gr gr */
```

TC

File Edit Run Compile Project Options Debug Break/watch

Edit

Error: Initializer syntax error in function main

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a[4]={ }, i;
clrscr();
printf("Elements ");for(i=0;i<4;i++)printf("%6d",a[i]);
getch();
}
/* Error */
```

