

## **POINTERS**

Pointer is a variable, which holds the address of another variable of same type.

Pointer is a memory location, which holds the address of another memory location.

Pointer is a derived data type.

### **Advantages:**

1. Dynamic memory allocation.
2. Program performance is increased due to preventing memory wastage.
3. They are very much used in System programming.
4. They are very much used in dynamic linked list & Stacks [data structures].
5. It allows to access a local variable outside the function i.e. data sharing between functions. [ call by address ].
6. To handle strings, arrays etc in functions we need pointers.
7. To handle data files we are using pointers.

8. They directly works on variable address. Due to this search time is reduced and execution speed is increased.

### **Disadvantage:**

They are not secured and make programmer complex.

### **Syntax:**

**datatype \* variable;**

- \* indicates it is a pointer data type.
- \* is called indirection operator.
- \* is called dereferencing operator.
- **\* is a re-direction operator.**
- \* indicates value at that address.
- \* indicates pointer value.

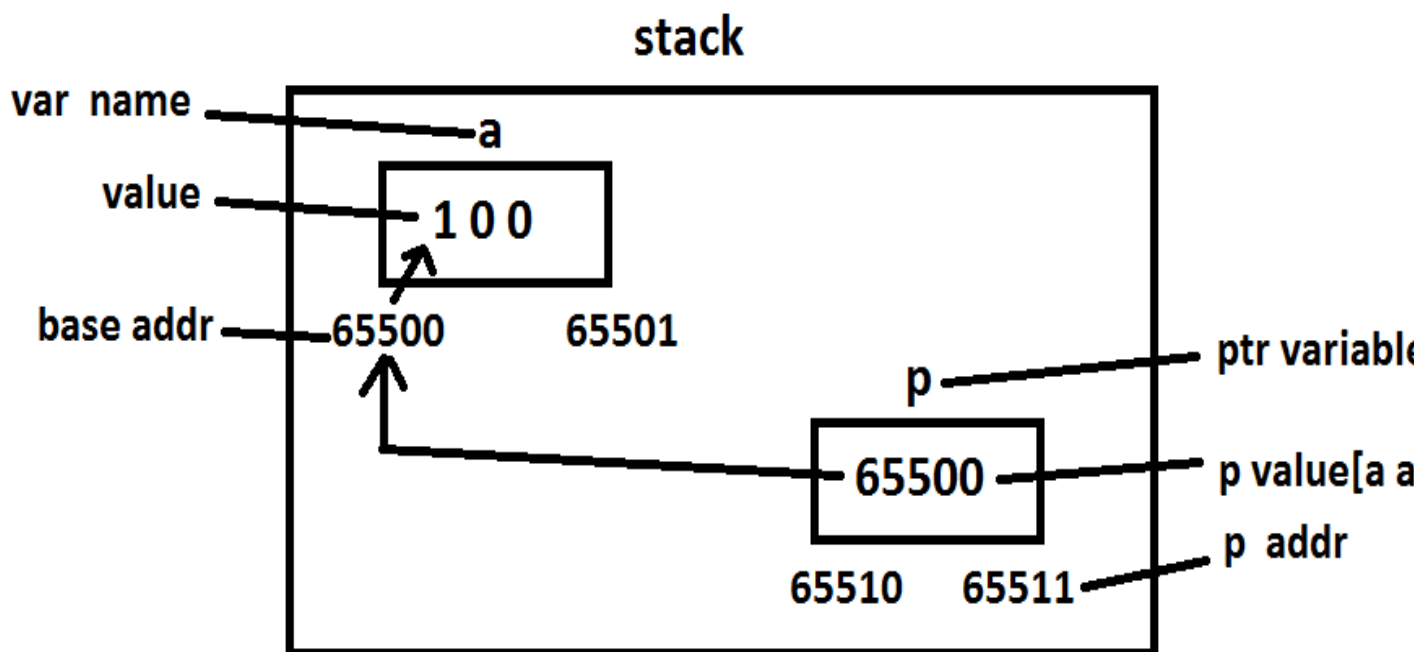
**Eg:**

```
int a=100, * p;
```

In the above example '**a**' is a general variable.

\* indicates '**p**' is a pointer type variable and it is able to store the address of general variable '**a**' as follows.

```
p = &a;
```



In the above example, to pick the value of **a** through pointer variable **p**, we have to use the **printf()** as follows.

`printf( "%d", *p );` → **100**

Here **\*p** means **value of p** or **value at that addr.** i.e. **65500**. But **65500** is the addr of **'a'**. The **value in a address is 100**.

Or

Here **p** means **65500**. **\*p** means value at **65500**. i.e. **100**.

Due to this example any changes we have conducted in **\*p** effects the value of **'a'**. Hence **p** is called **pointer** to **a**.

Eg: **\*p=200;**

Now a becomes 200.

**Eg:**

**Finding a variable value and address  
using a pointer:**

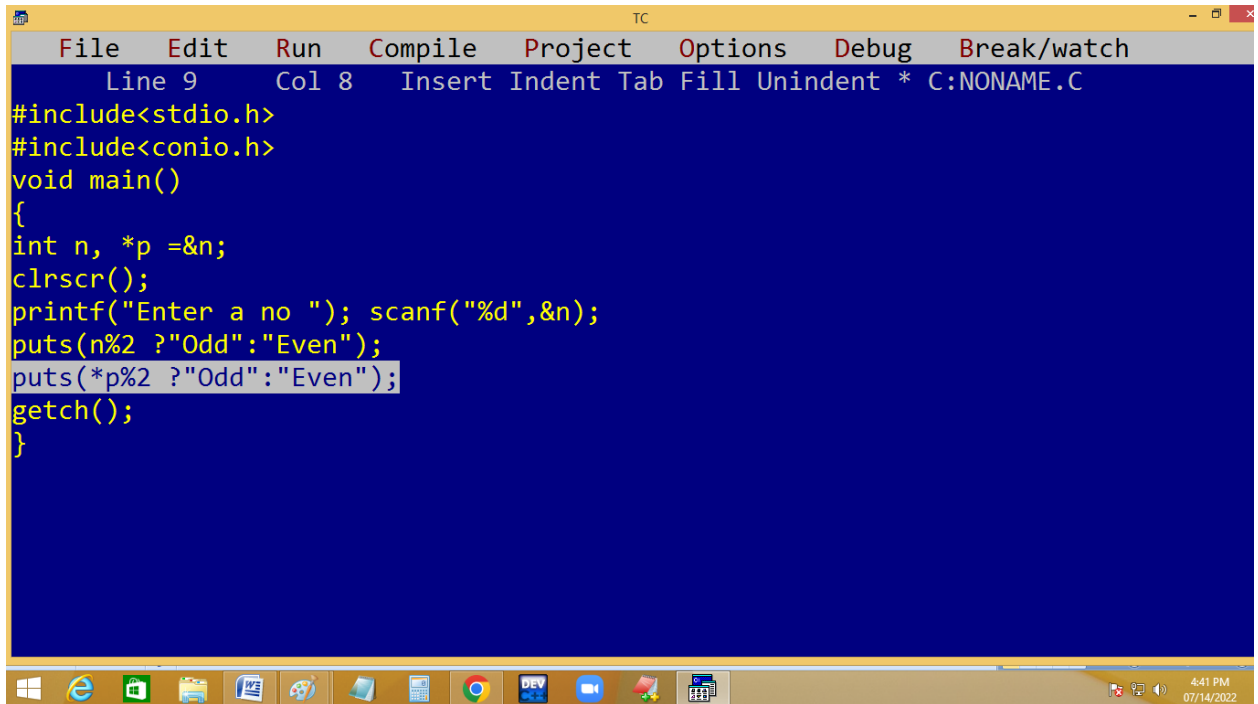
```
TC
Line 1 Col 2 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a=10, *p;
p=&a;
clrscr();
printf("a=%d\n",a);
printf("a addr  =%u\n",&a);
printf("p value  =%u\n",p);
printf("a value through p = %d\n",*p);
*p=20;
printf("a=%d, *p=%d\n",a,*p);
a=30;
printf("a=%d, *p=%d\n",a,*p);
getch();
}
```

```
TC
a=10
a addr  =65498
p value  =65498
a value through p = 10
a=20, *p=20
a=30, *p=30

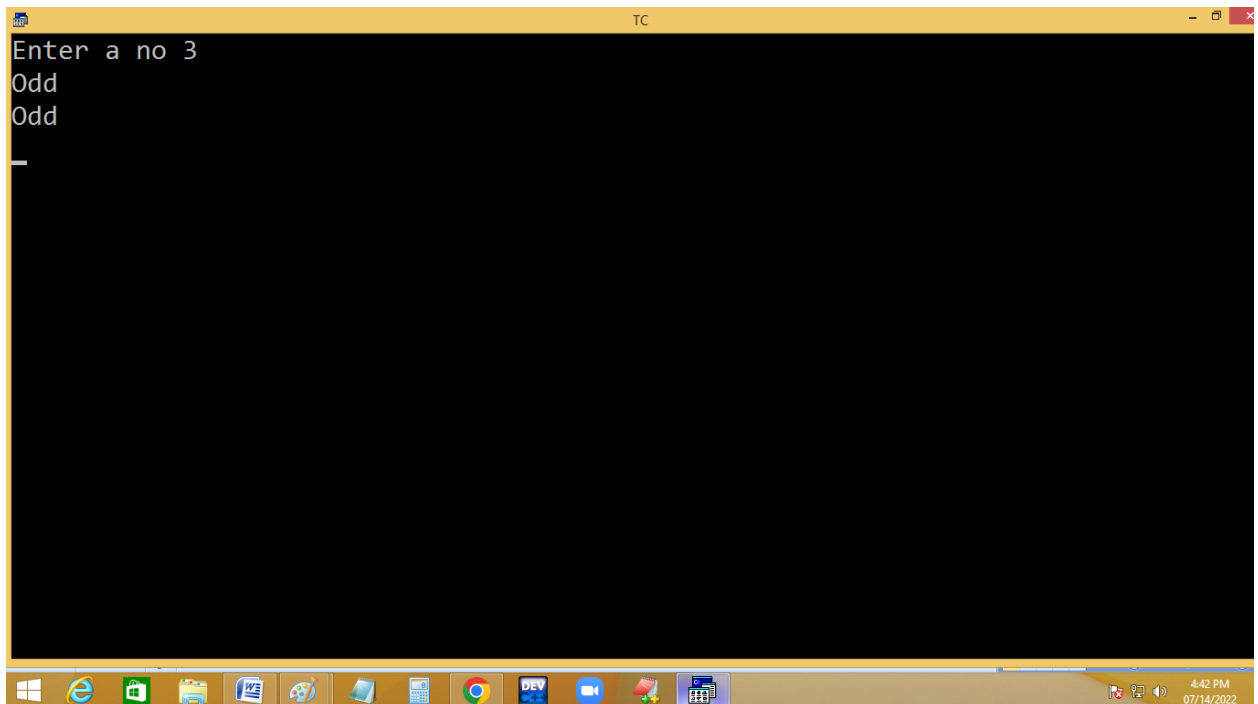
```

Page: 5 of 5 Words: 327 100%

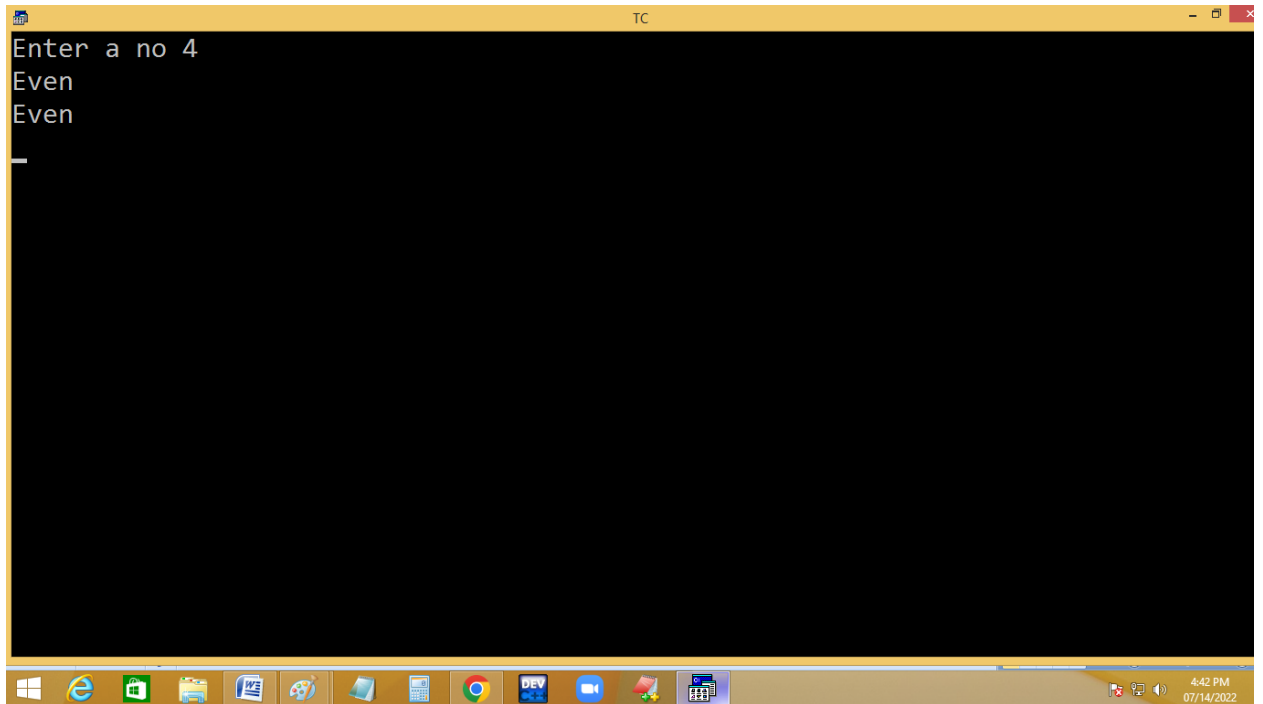
## Eg. finding even/odd using pointer.



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 8 Insert Indent Tab Fill Unindent * C:\NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, *p =&n;
clrscr();
printf("Enter a no "); scanf("%d",&n);
puts(n%2 ? "Odd": "Even");
puts(*p%2 ? "Odd": "Even");
getch();
}
```



```
TC
Enter a no 3
Odd
Odd
```

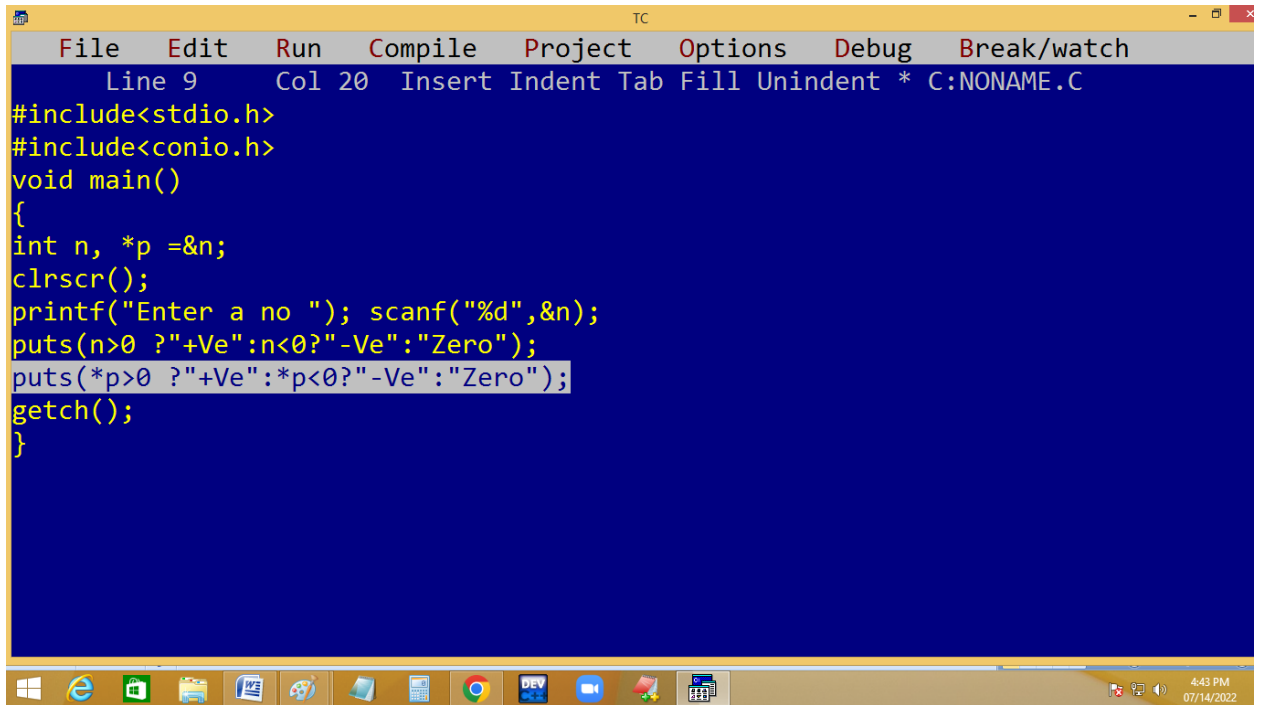


```
TC
Enter a no 4
Even
Even
_
```

The screenshot shows a Turbo C++ (TC) window with a black background. The text "Enter a no 4" is displayed on the first line, followed by "Even" on the second and third lines. A cursor is visible on the fourth line. The Windows taskbar is visible at the bottom, showing various application icons and the system clock indicating 4:42 PM on 07/14/2022.

**Eg. Finding +ve/-ve/0 using pointer.**

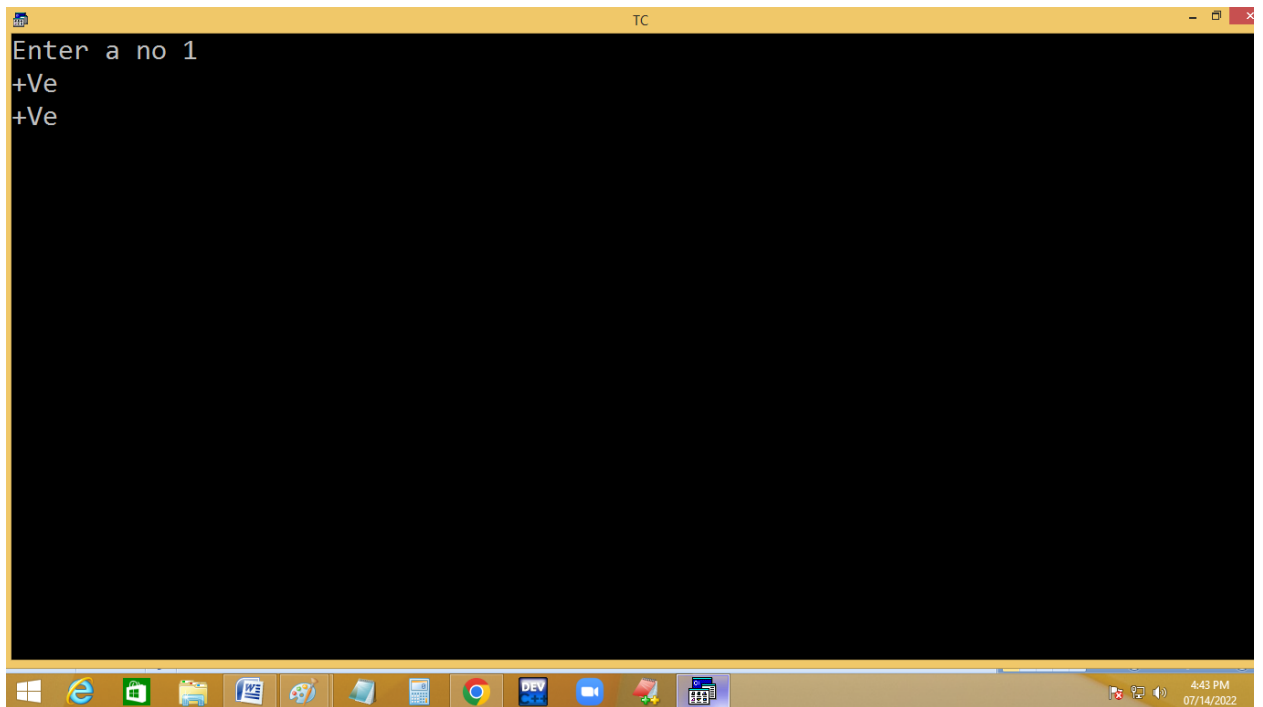




The screenshot shows the Turbo C++ (TC) IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a toolbar. The main window displays the source code for a C program named C:\NONAME.C. The code is as follows:

```
Line 9 Col 20 Insert Indent Tab Fill Unindent * C:\NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, *p =&n;
clrscr();
printf("Enter a no "); scanf("%d",&n);
puts(n>0 ? "+Ve":n<0?"-Ve":"Zero");
puts(*p>0 ? "+Ve":*p<0?"-Ve":"Zero");
getch();
}
```

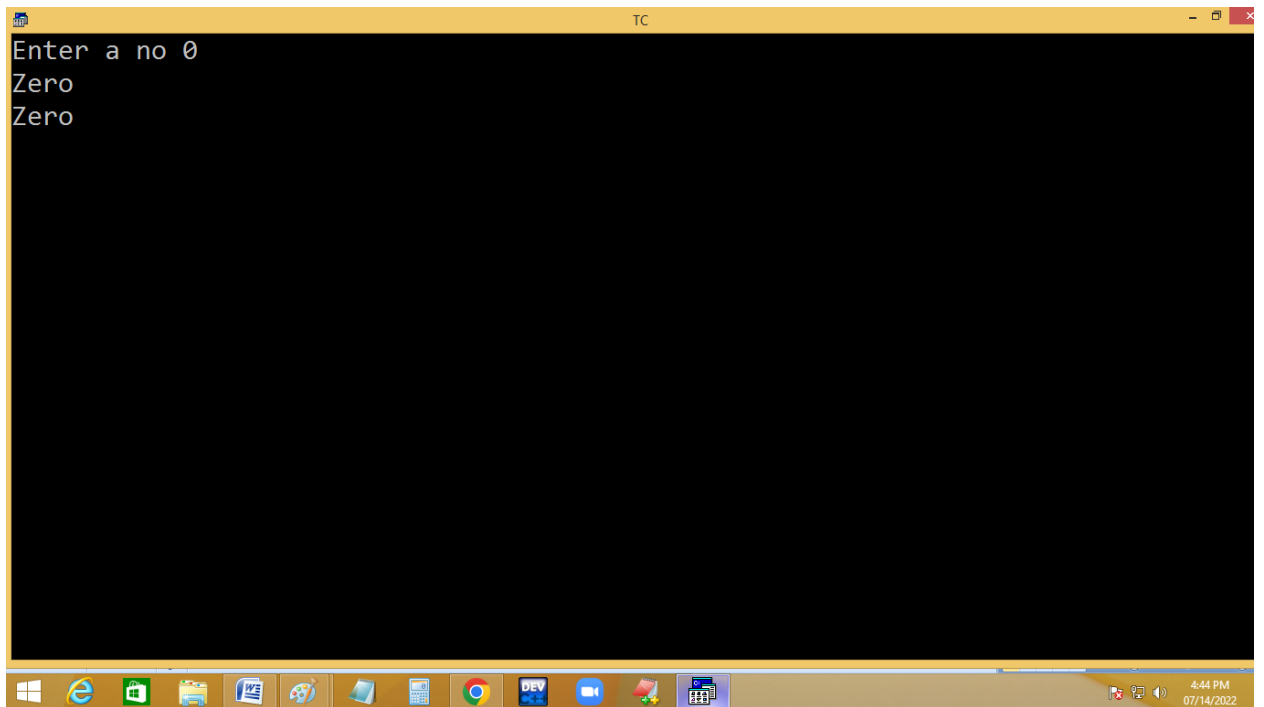
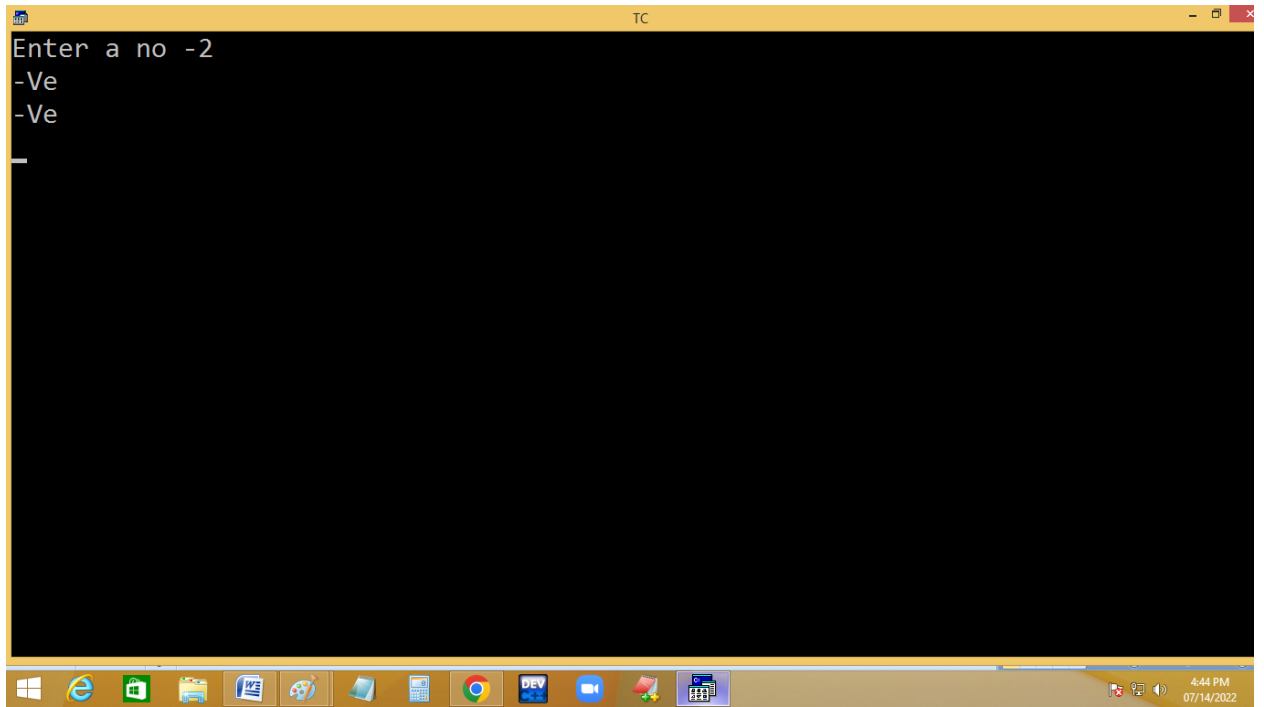
The taskbar at the bottom shows various application icons and the system clock indicating 4:43 PM on 07/14/2022.



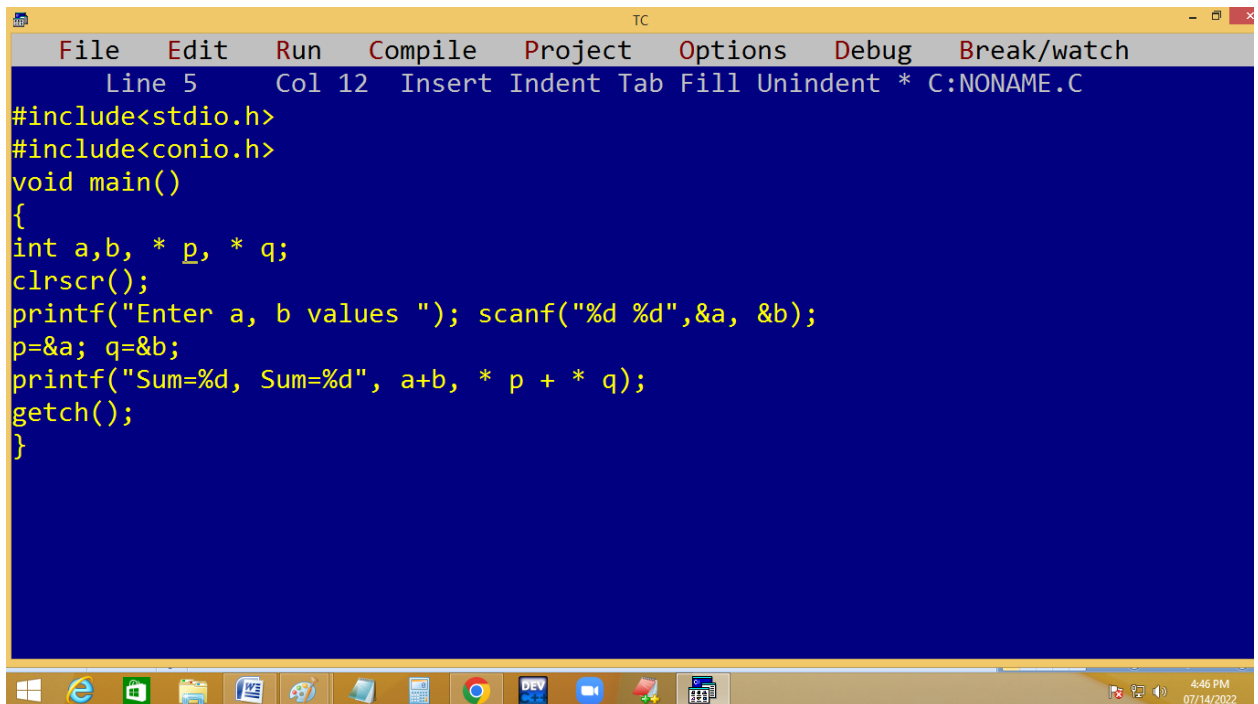
The screenshot shows the Turbo C++ (TC) IDE after the program has been executed. The main window displays the output of the program:

```
Enter a no 1
+Ve
+Ve
```

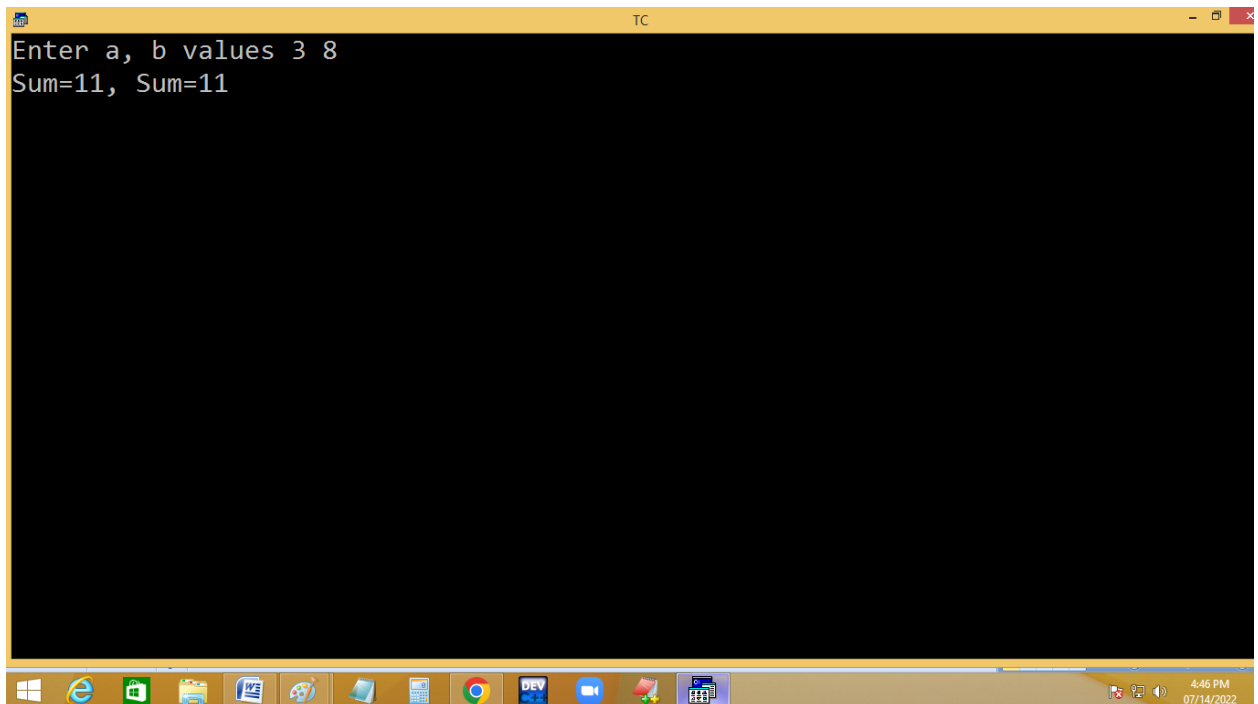
The taskbar at the bottom remains the same, showing the system clock at 4:43 PM on 07/14/2022.



## Eg. Add two numbers using pointers.

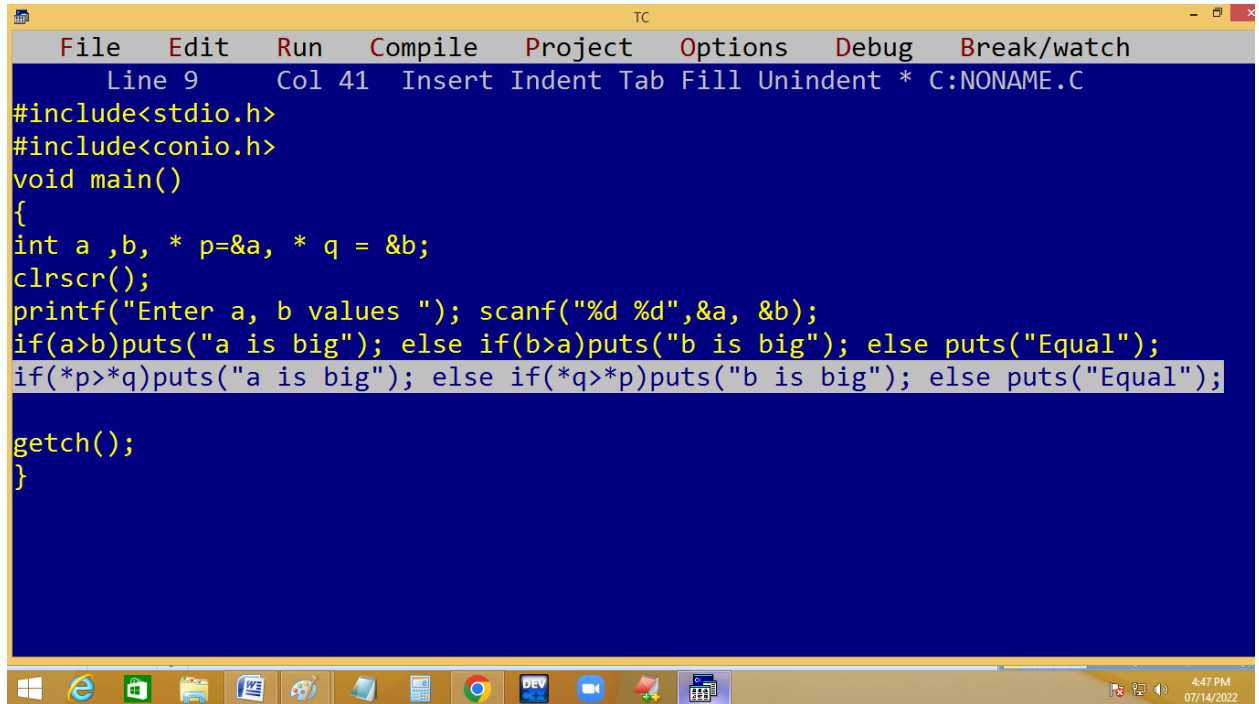


```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 12 Insert Indent Tab Fill Unindent * C:\NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b, * p, * q;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
p=&a; q=&b;
printf("Sum=%d, Sum=%d", a+b, * p + * q);
getch();
}
```



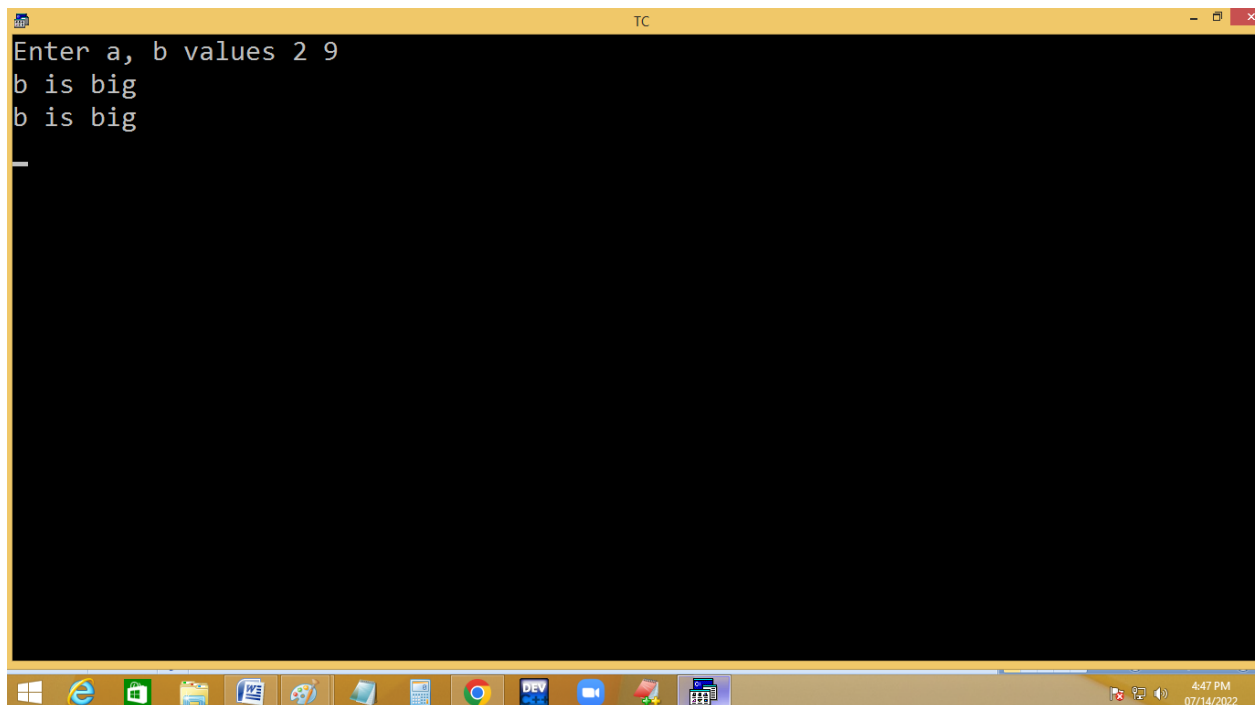
```
TC
Enter a, b values 3 8
Sum=11, Sum=11
```

**Eg. find max in 2 numbers using pointer.**

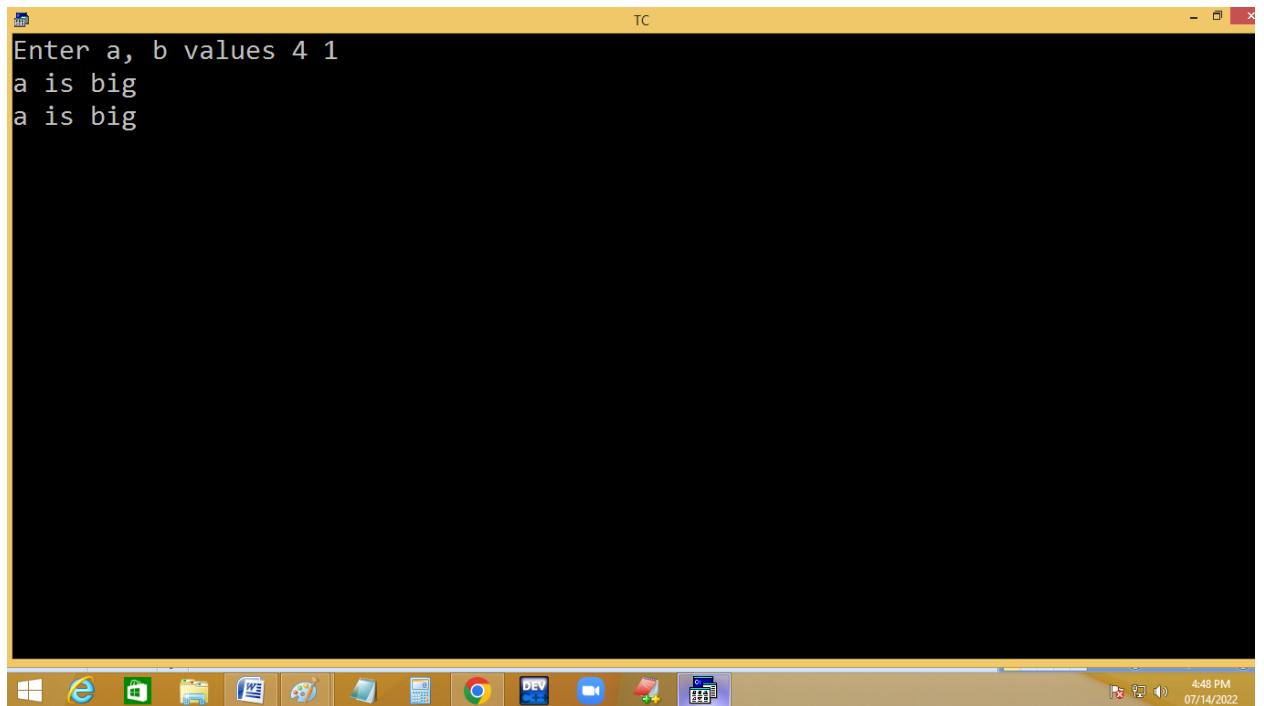


```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 41 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a ,b, * p=&a, * q = &b;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
if(a>b)puts("a is big"); else if(b>a)puts("b is big"); else puts("Equal");
if(*p>*q)puts("a is big"); else if(*q>*p)puts("b is big"); else puts("Equal");

getch();
}
```

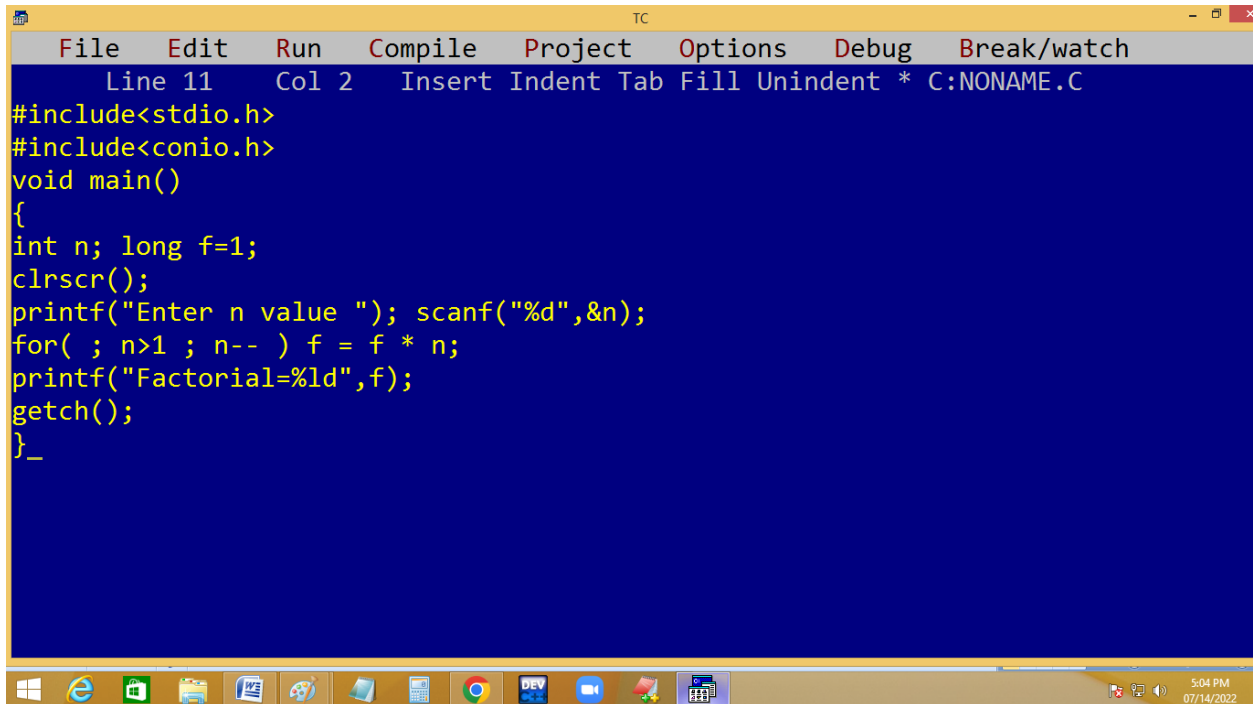


```
TC
Enter a, b values 2 9
b is big
b is big
_
```

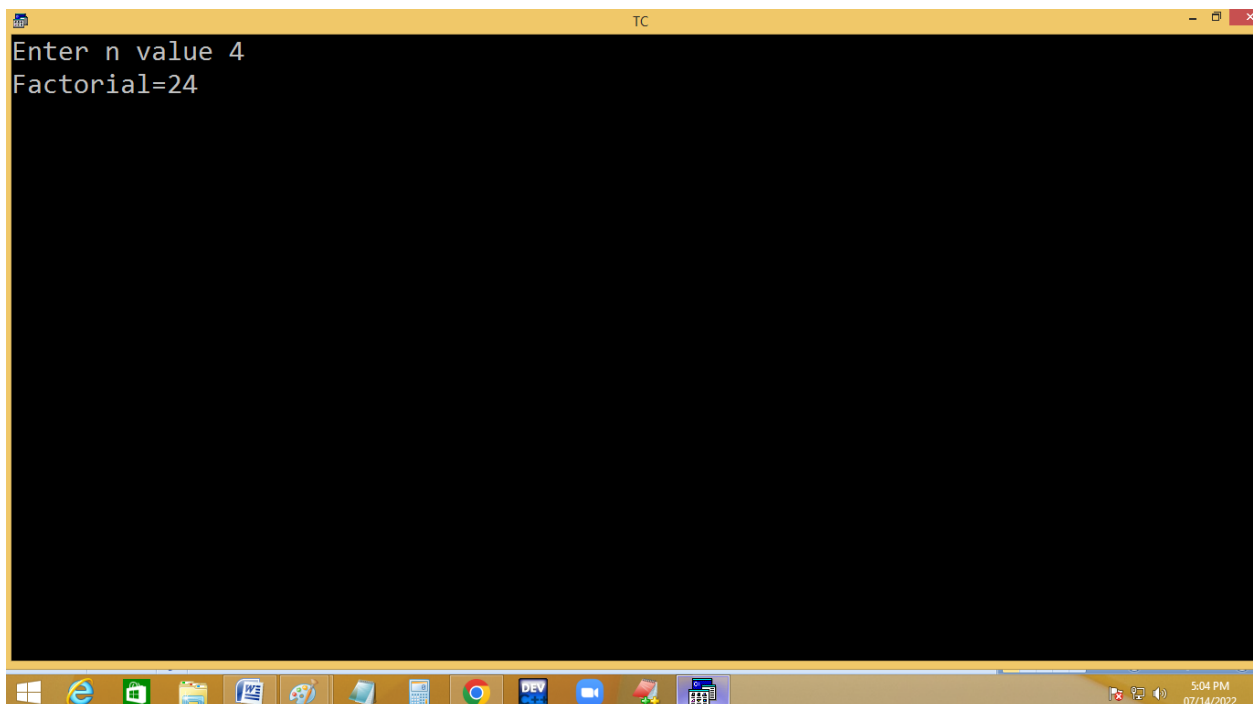


```
TC
Enter a, b values 4 1
a is big
a is big
```

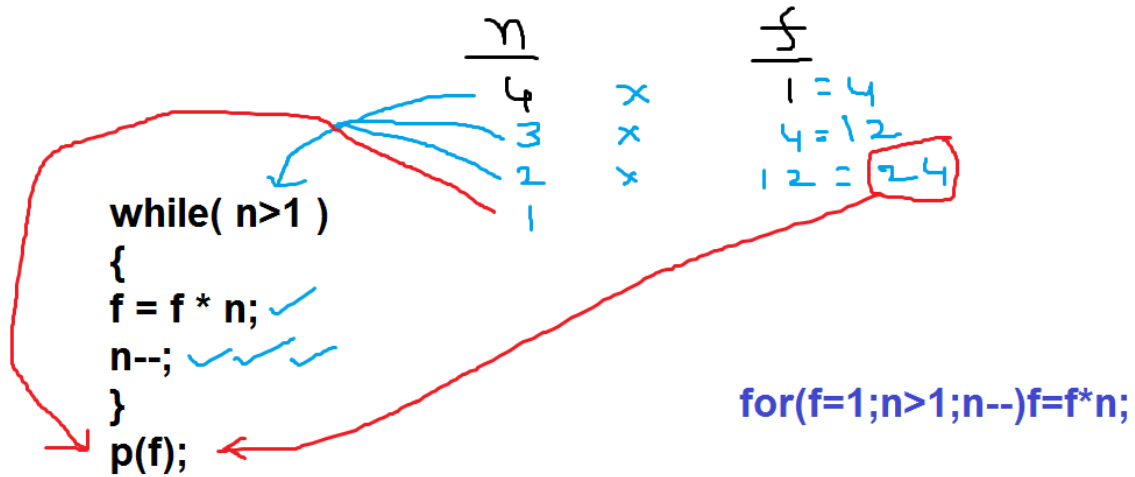
# Finding factorial using pointer.

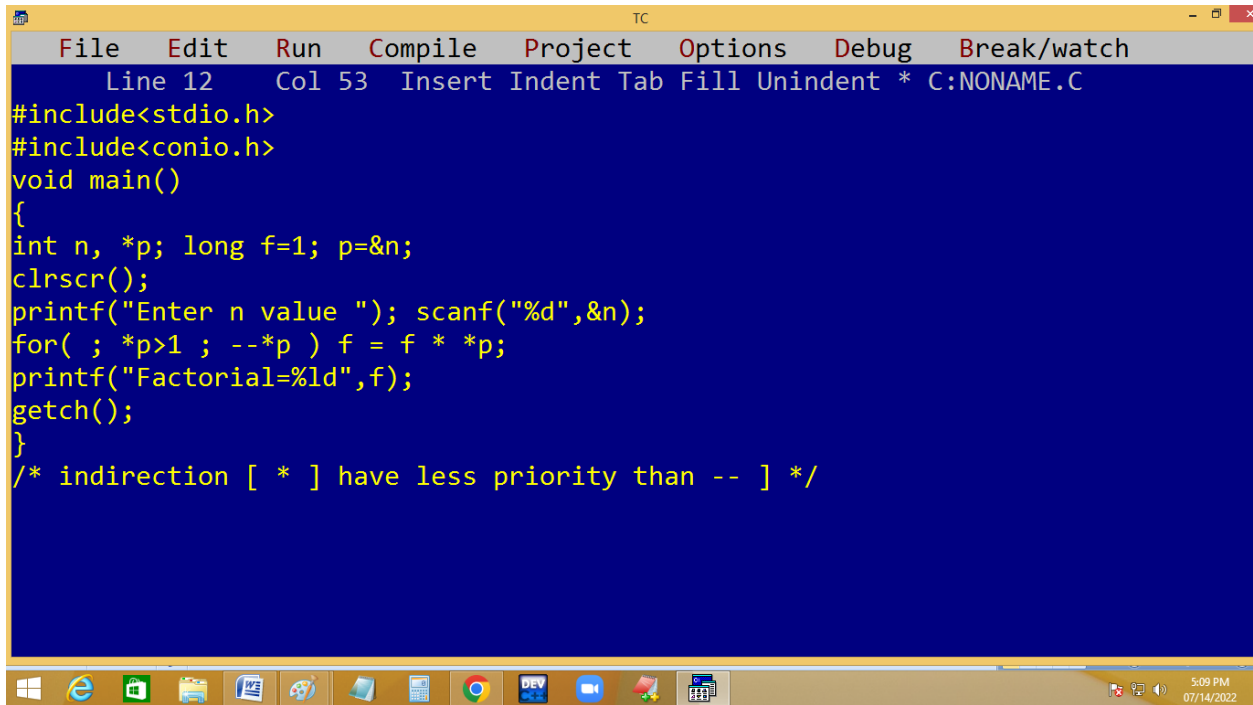


```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 2 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n; long f=1;
clrscr();
printf("Enter n value "); scanf("%d",&n);
for( ; n>1 ; n-- ) f = f * n;
printf("Factorial=%ld",f);
getch();
}_
```



```
TC
Enter n value 4
Factorial=24
```

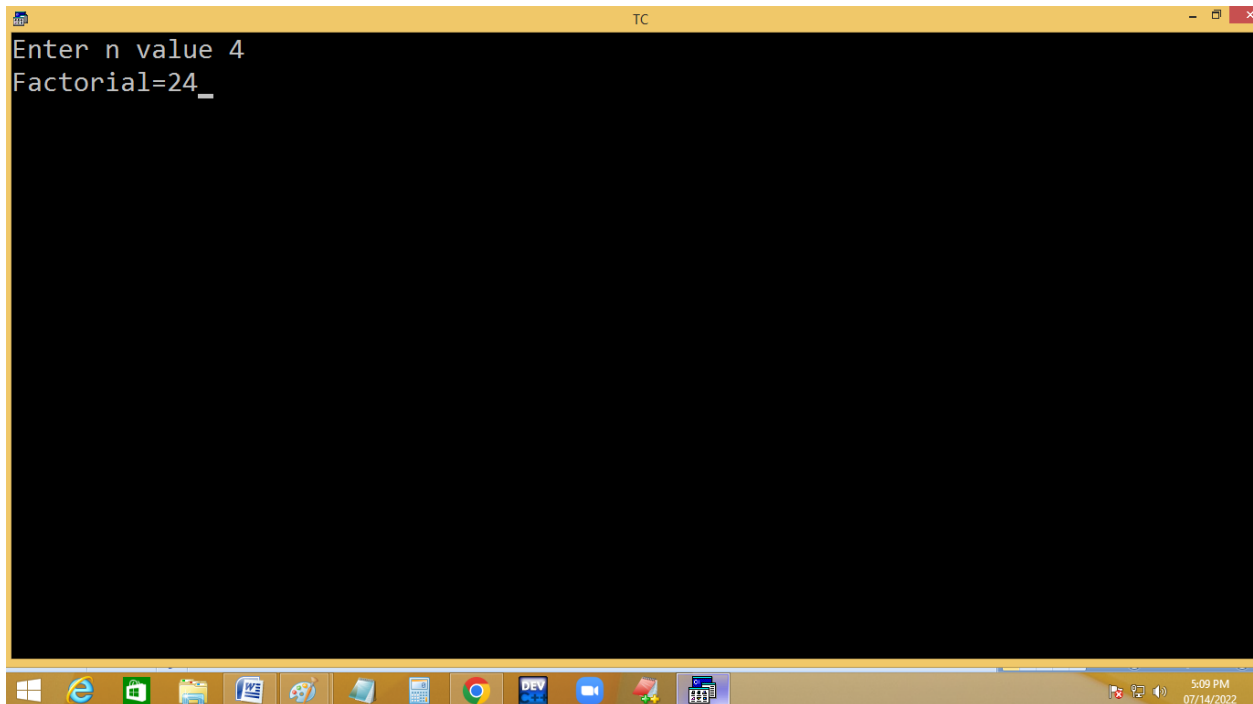




The screenshot shows the Turbo C++ (TC) IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a toolbar. The code editor displays the following C program:

```
Line 12 Col 53 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, *p; long f=1; p=&n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
for( ; *p>1 ; --*p ) f = f * *p;
printf("Factorial=%ld",f);
getch();
}
/* indirection [ * ] have less priority than -- ] */
```

The Windows taskbar at the bottom shows various application icons and the system clock indicating 5:09 PM on 07/14/2022.



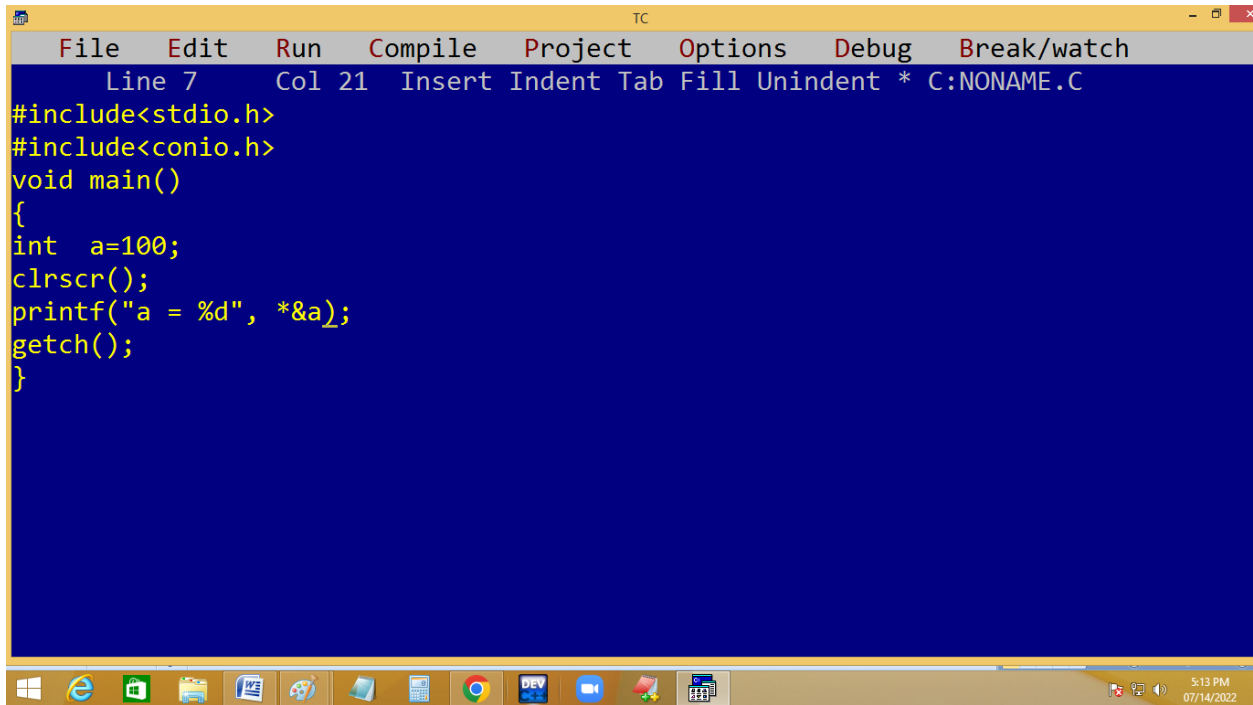
The screenshot shows the Turbo C++ (TC) IDE after execution. The output window displays the following text:

```
Enter n value 4
Factorial=24_
```

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

**Eg. printing a normal variable value  
using pointer mechanism.**

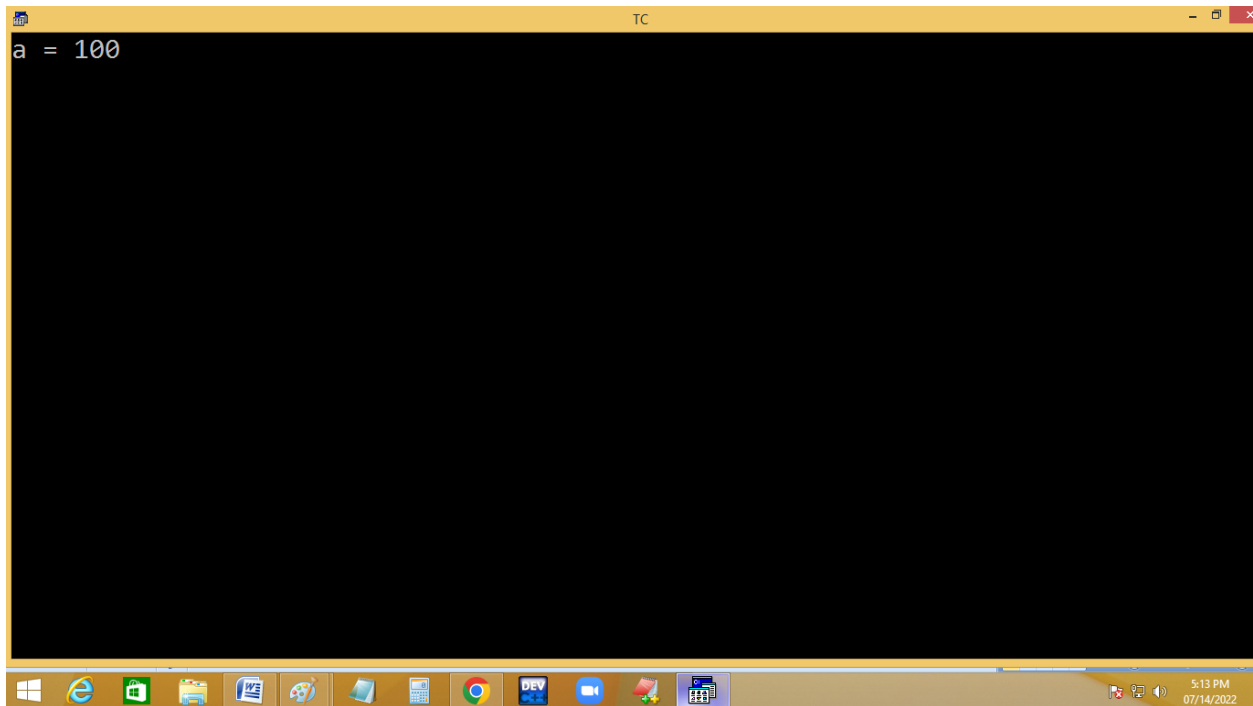




The screenshot shows the Turbo C++ (TC) IDE with a yellow title bar and a menu bar containing File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 7 Col 21 Insert Indent Tab Fill Unindent \* C:NONAME.C'. The main editing area has a blue background and contains the following C code:

```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a=100;
clrscr();
printf("a = %d", *&a);
getch();
}
```

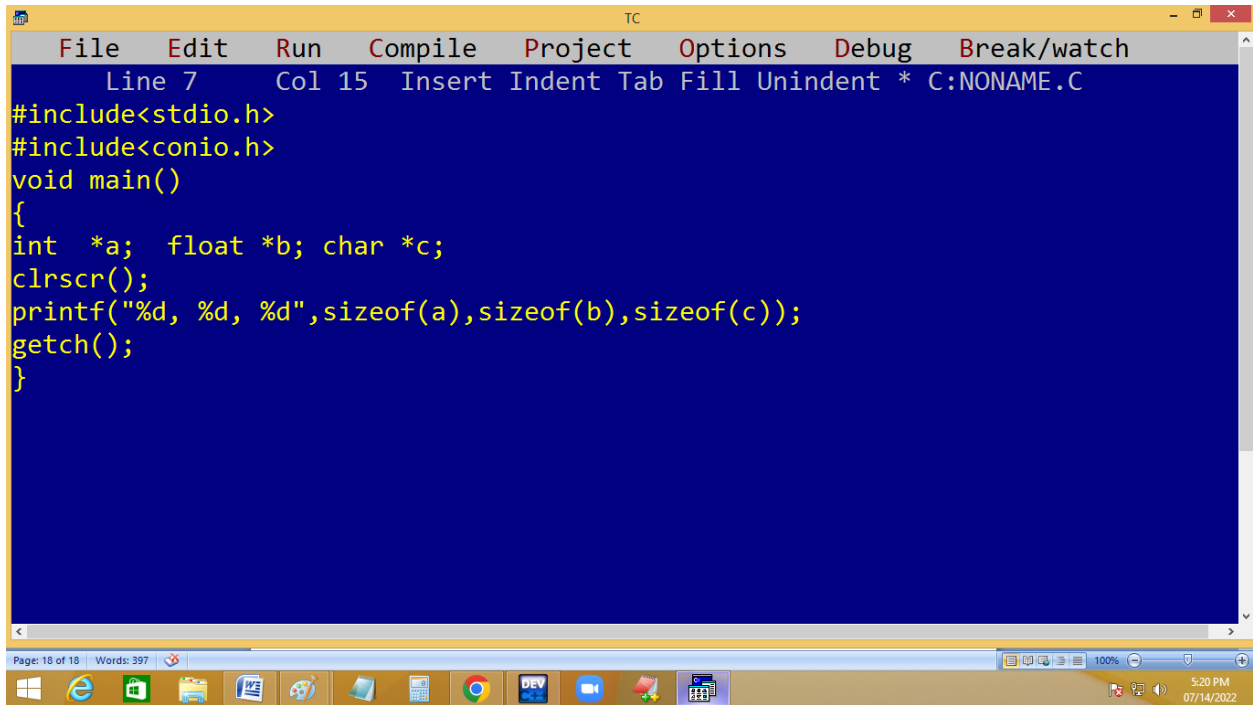
The Windows taskbar at the bottom shows various application icons and the system clock displaying 5:13 PM on 07/14/2022.



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: 'a = 100'. The Windows taskbar at the bottom is identical to the first screenshot, showing the same application icons and system clock (5:13 PM on 07/14/2022).

## **Eg. finding pointer size.**

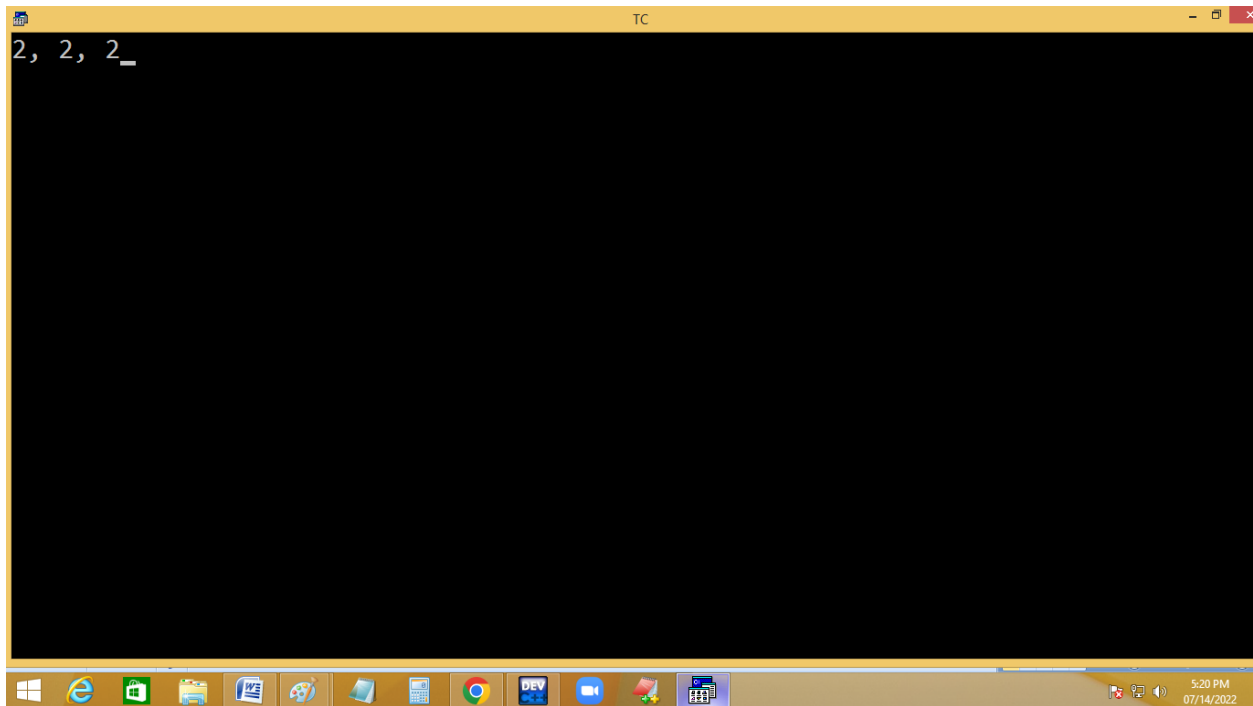
Pointer stores always the address and address is unsigned int. due to this pointer takes 2 / 4 / 8 bytes in 16 / 32 / 64 bit compilers.



The screenshot shows the Turbo C++ (TC) IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a toolbar. The main window has a blue background and displays the following C code:

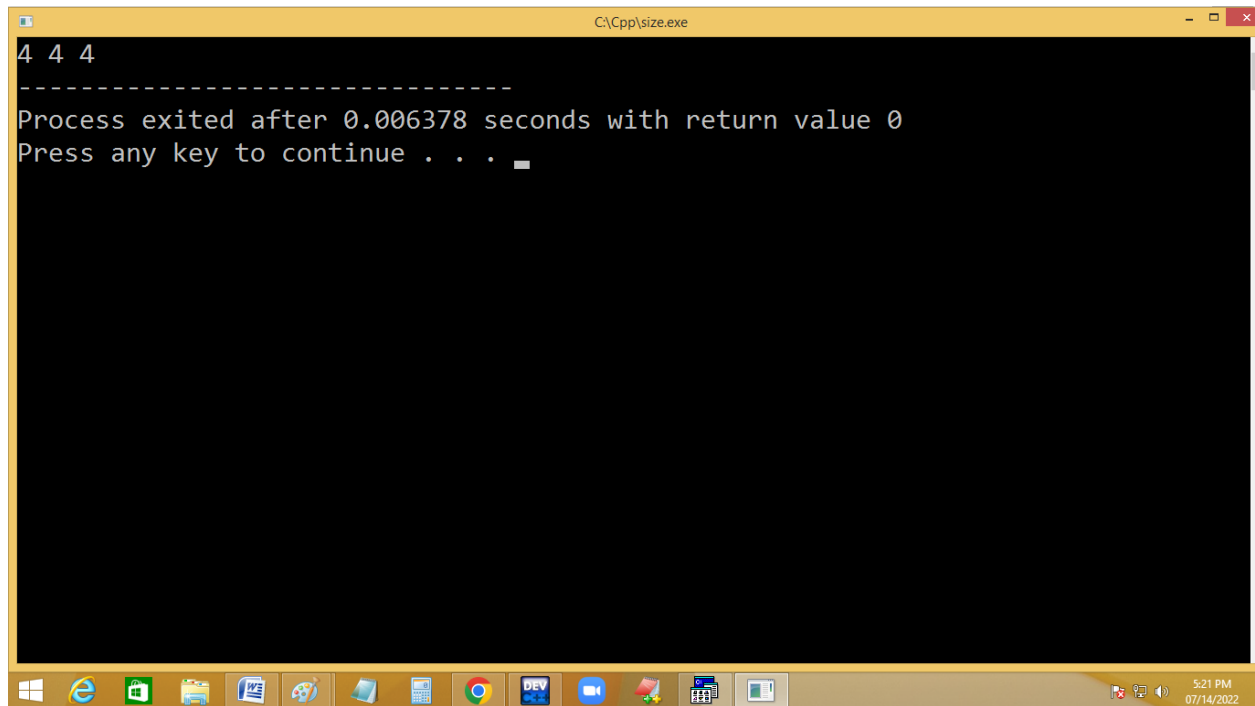
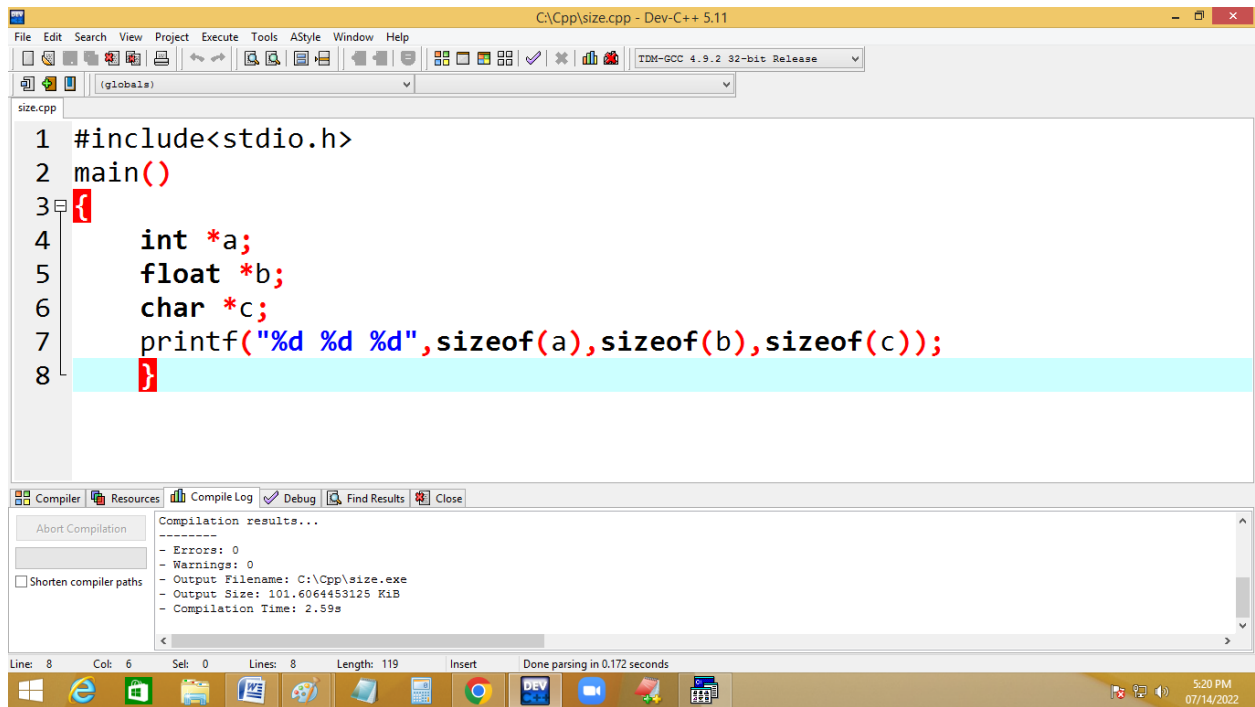
```
Line 7 Col 15 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int *a; float *b; char *c;
clrscr();
printf("%d, %d, %d",sizeof(a),sizeof(b),sizeof(c));
getch();
}
```

The status bar at the bottom indicates "Page: 18 of 18" and "Words: 397". The Windows taskbar at the bottom shows various application icons and the system clock displaying "5:20 PM 07/14/2022".



The screenshot shows the Turbo C++ (TC) IDE with a black background. The output of the program is displayed in the top-left corner of the main window as "2, 2, 2\_". The status bar at the bottom shows "Page: 18 of 18" and "Words: 397". The Windows taskbar at the bottom shows various application icons and the system clock displaying "5:20 PM 07/14/2022".

**In dev c++:**



# online compiler:

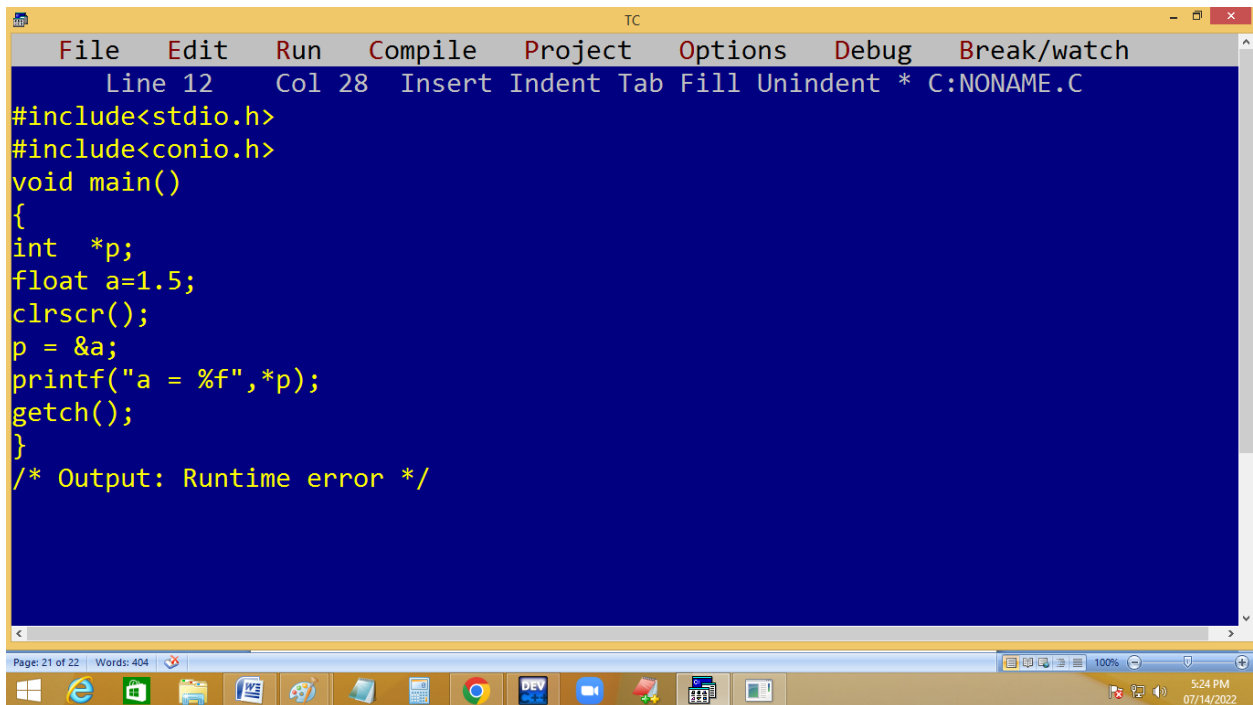
```
main.c
1 // Online C compiler to run C program online
2 #include<stdio.h>
3 main()
4 {
5     int *a;
6     float *b;
7     char *c;
8     printf("%d %d %d", sizeof(a), sizeof(b), sizeof(c));
9 }
```

Output

```
/tmp/xSRXuk4gLG.o
8 8 8
```

Introducing the most interactive C Course. Learn practically and get certified. Get Started!

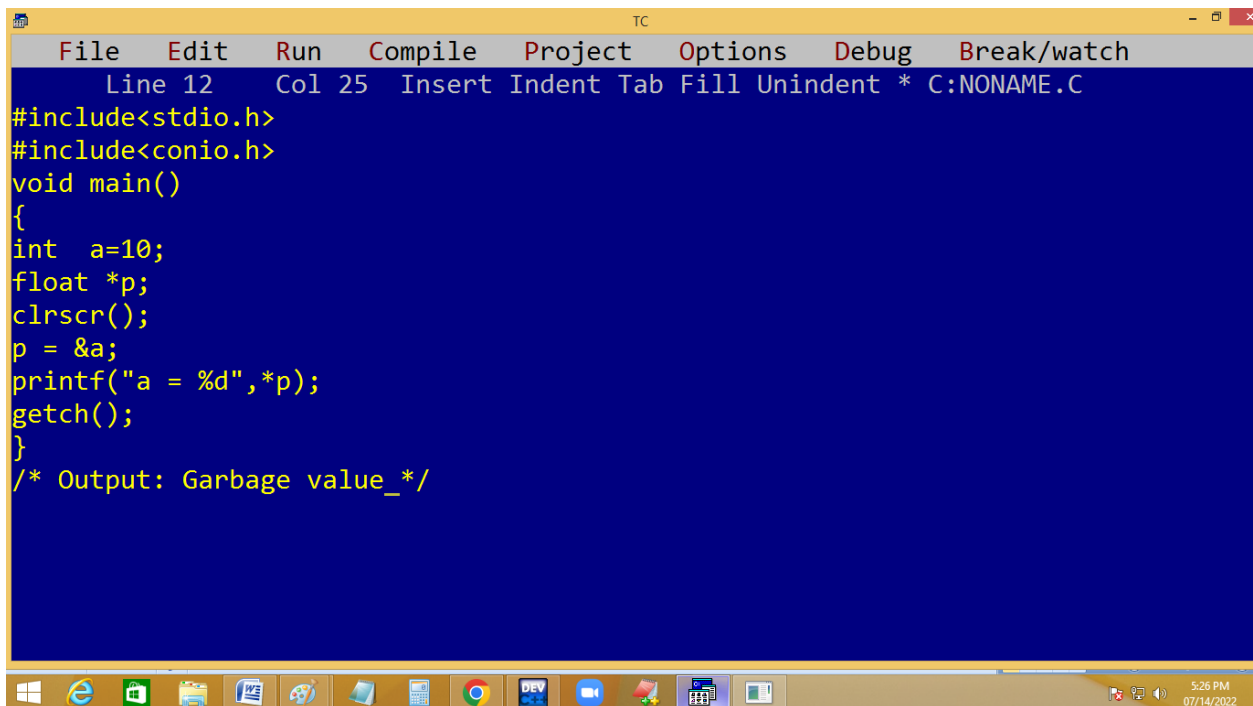
**Pointer compatibility:** Pointer stores address of same type of variable. When different type is given, it gives garbage or runtime error. They are used to handle dynamic multi dimensional array.



The screenshot shows the Turbo C++ IDE with a C program. The code is as follows:

```
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 28 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int *p;
float a=1.5;
clrscr();
p = &a;
printf("a = %f",*p);
getch();
}
/* Output: Runtime error */
```

The status bar at the bottom indicates "Page: 21 of 22" and "Words: 404". The Windows taskbar at the bottom shows the time as 5:24 PM on 07/14/2022.

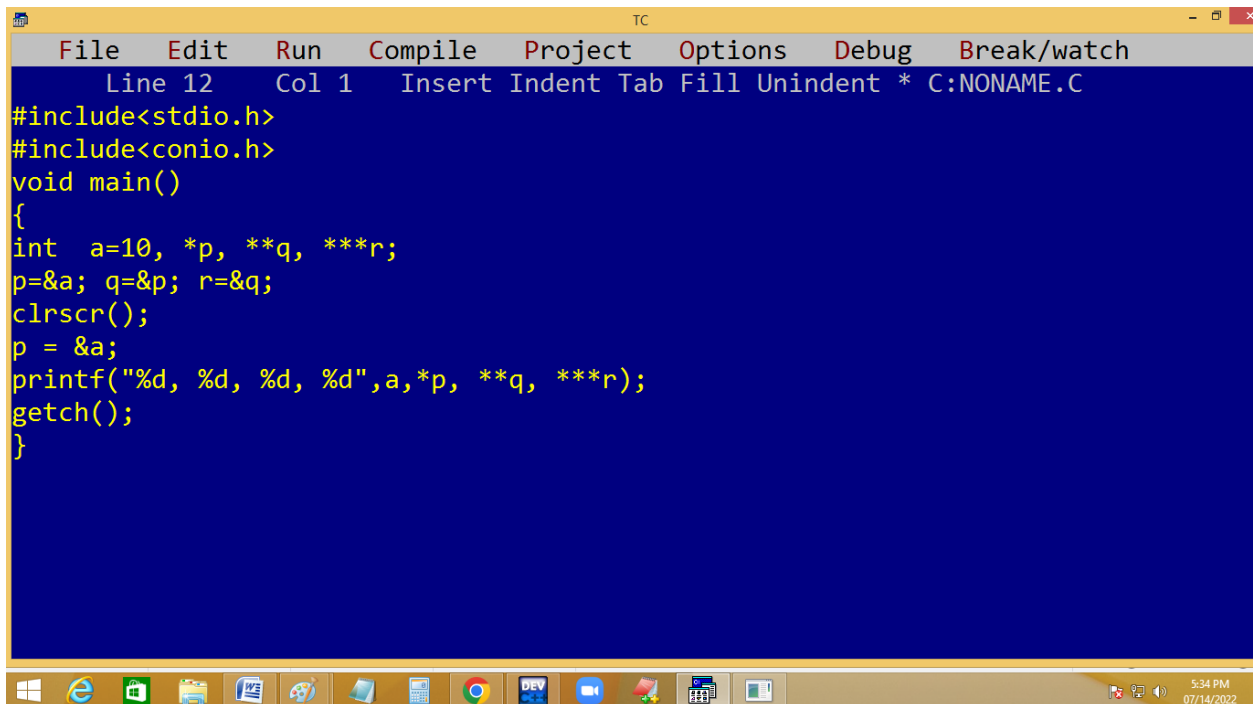


The screenshot shows the Turbo C++ IDE with a C program. The code is as follows:

```
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 25 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a=10;
float *p;
clrscr();
p = &a;
printf("a = %d",*p);
getch();
}
/* Output: Garbage value_*/
```

The status bar at the bottom indicates "Page: 21 of 22" and "Words: 404". The Windows taskbar at the bottom shows the time as 5:26 PM on 07/14/2022.

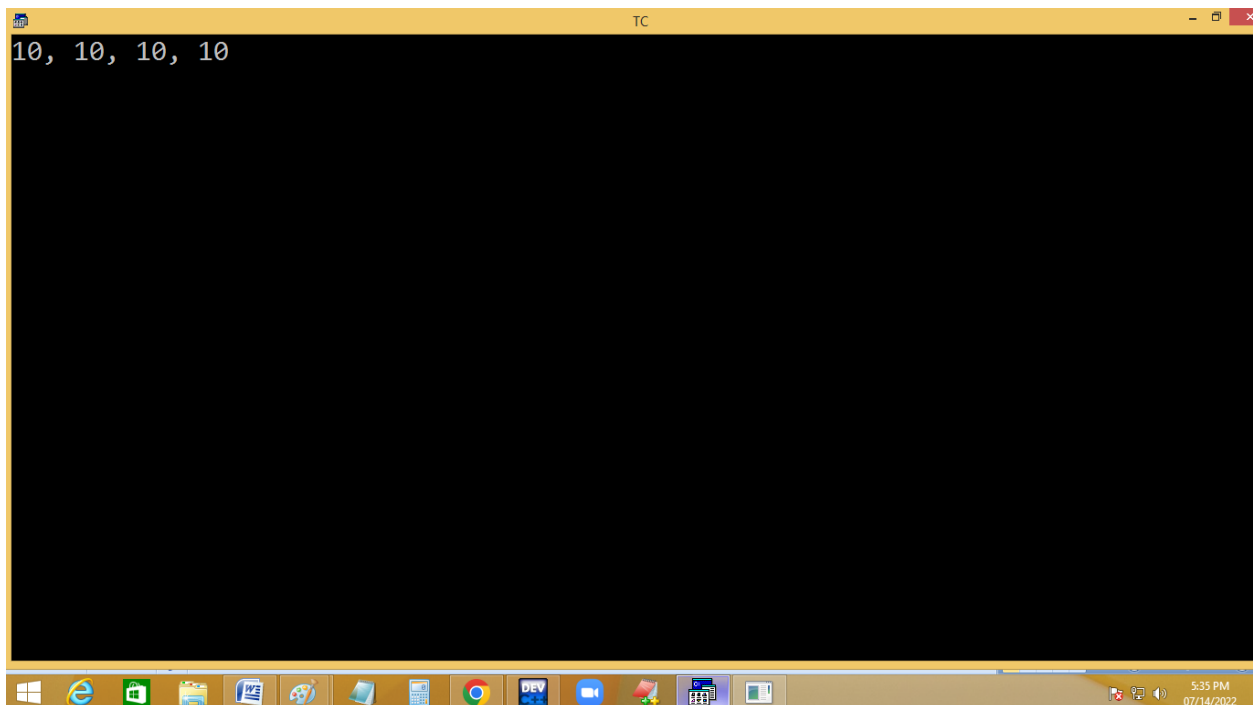
# Pointer to pointer / double pointer:



The screenshot shows the Turbo C++ (TC) IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a status bar (Line 12, Col 1, Insert, Indent, Tab, Fill, Unindent, \* C:NONAME.C). The code in the editor is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
int  a=10, *p, **q, ***r;
p=&a; q=&p; r=&q;
clrscr();
p = &a;
printf("%d, %d, %d, %d",a,*p, **q, ***r);
getch();
}
```

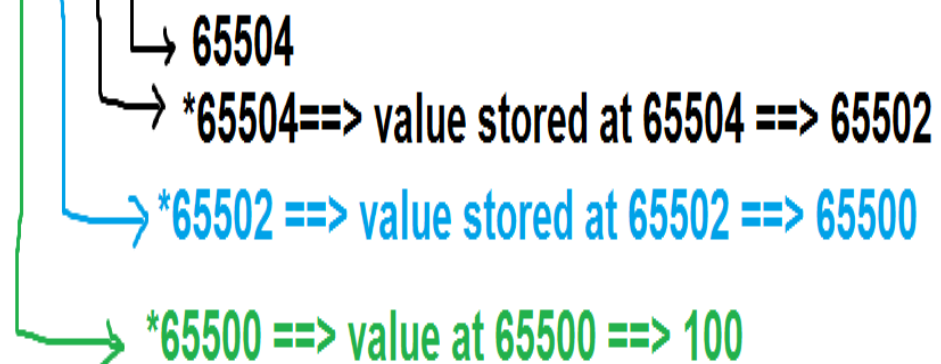
The Windows taskbar at the bottom shows the time as 5:34 PM on 07/14/2022.



The screenshot shows the Turbo C++ (TC) IDE with the output of the program displayed in the main window: "10, 10, 10, 10". The Windows taskbar at the bottom shows the time as 5:35 PM on 07/14/2022.

stack		
variable	value	addr
r	65504	65506
q	65502	65504
p	65500	65502
a	100	65500

`p(" %d ", ***r);`

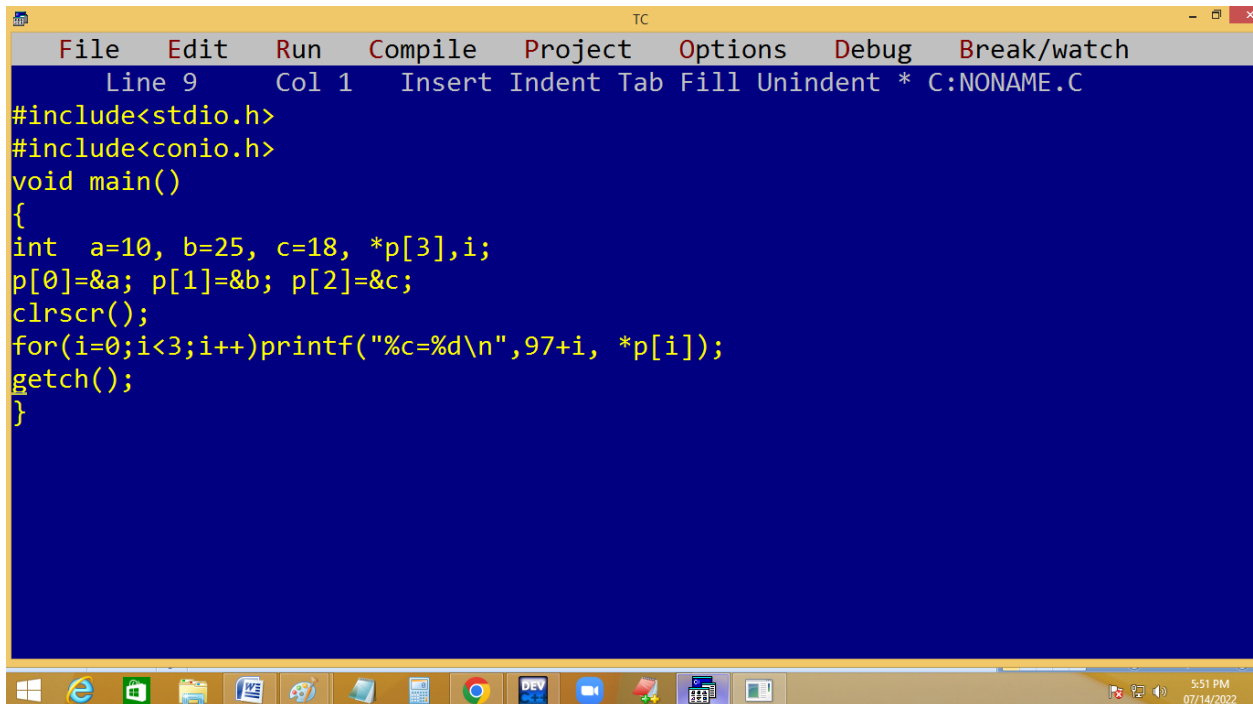


## Array of pointer:

Like general variables we can declare the pointer using array. Due to this



**we can store several address in one pointer variable. They are used to control dynamic array.**



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 1 Insert Indent Tab Fill Unindent * C:NONAME.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a=10, b=25, c=18, *p[3],i;
p[0]=&a; p[1]=&b; p[2]=&c;
clrscr();
for(i=0;i<3;i++)printf("%c=%d\n",97+i, *p[i]);
getch();
}
```

The screenshot shows a Turbo C++ IDE window titled 'TC'. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The status bar at the top indicates 'Line 9 Col 1 Insert Indent Tab Fill Unindent \* C:NONAME.C'. The code in the editor is a C program that declares an array of pointers, stores addresses of variables a, b, and c, and prints the values pointed to by each element of the array. The Windows taskbar at the bottom shows various icons and the system clock indicating 5:51 PM on 07/14/2022.

```
TC
a=10
b=25
c=18
```

stack

