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 local variable outside the function i.e. data sharing between functions.** [**call by address/Reference**].
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.

Dis-advantage:

They are not secured.

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.

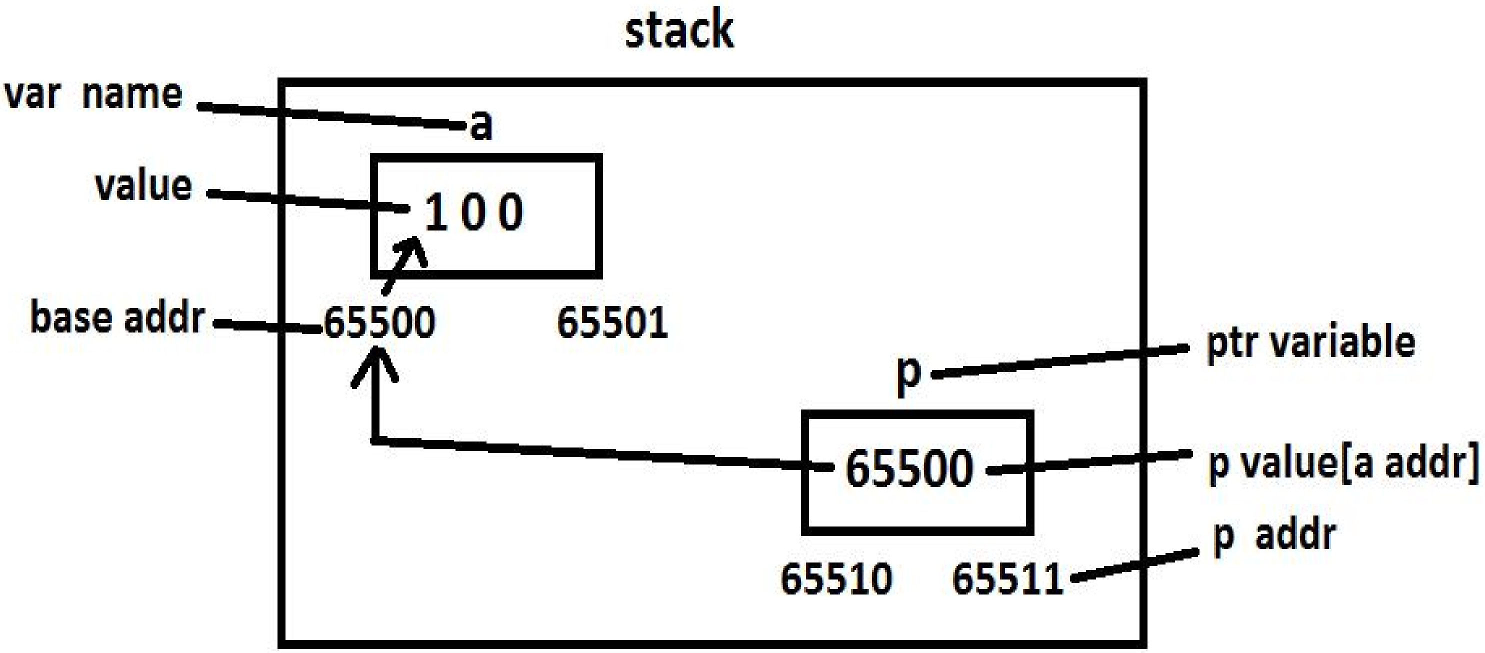
Eg:

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.

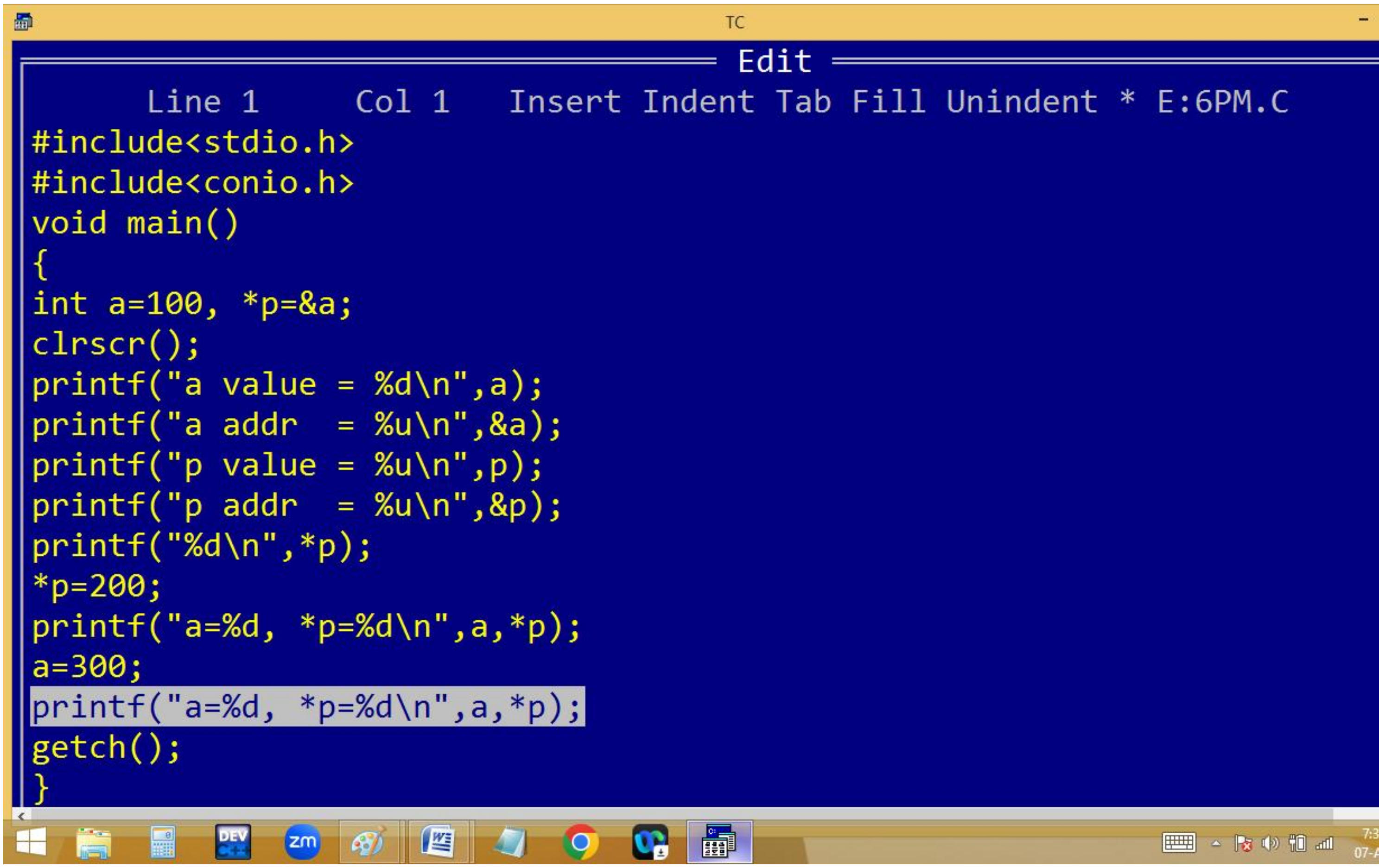
Due to this example any changes conducted in ***p** effects the value of **‘a’**. Hence **p** is called **pointer** to **a**.

Eg: *p=200;

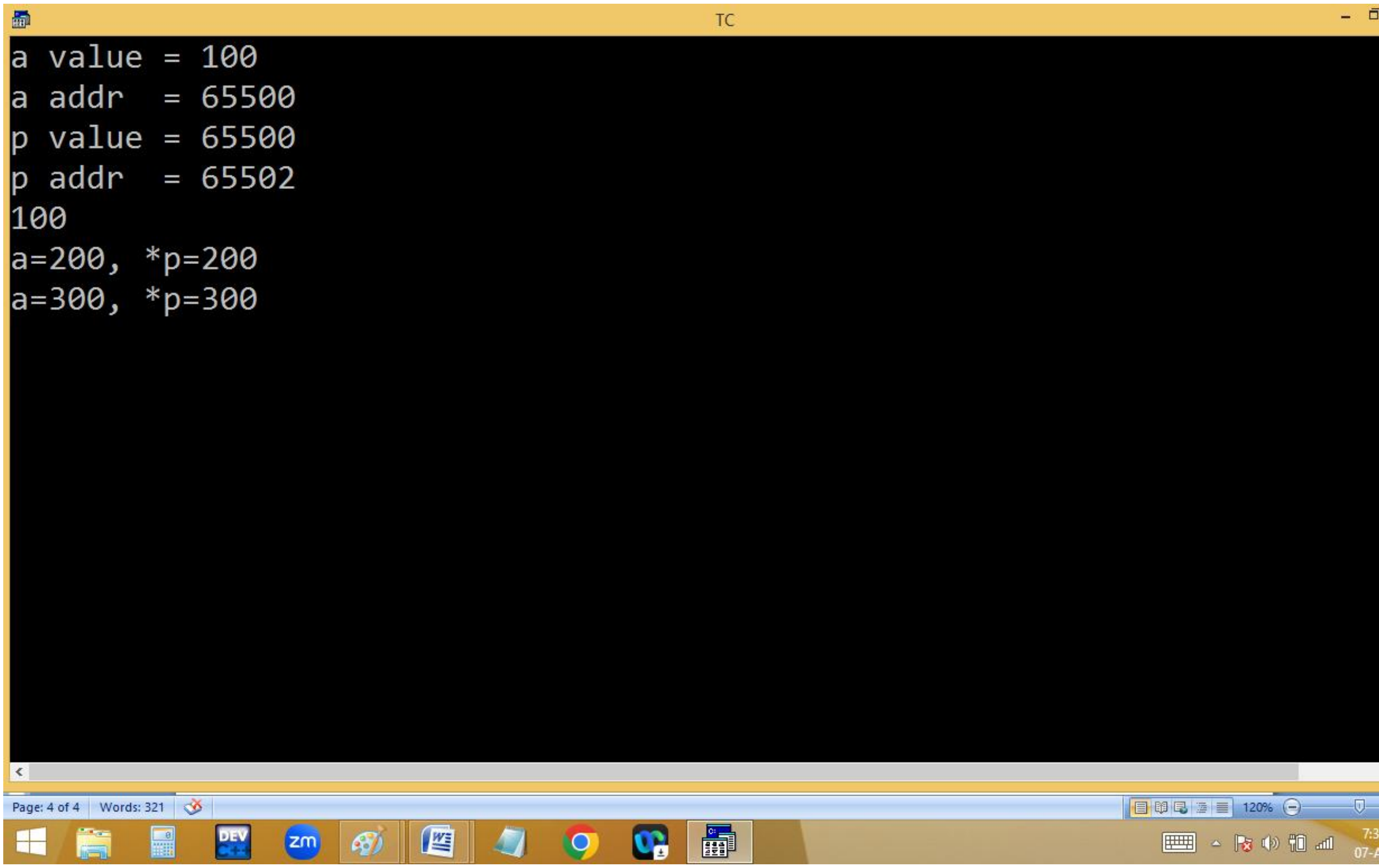
Now **a** value becomes **200**.

Eg:

Finding a variable value and address using a pointer:



```
      Line 1     Col 1   Insert Indent Tab Fill Unindent * E:6PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a=100, *p=&a;
clrscr();
printf("a value = %d\n",a);
printf("a addr  = %u\n",&a);
printf("p value = %u\n",p);
printf("p addr  = %u\n",&p);
printf("%d\n",*p);
*p=200;
printf("a=%d, *p=%d\n",a,*p);
a=300;
printf("a=%d, *p=%d\n",a,*p);
getch();
}
```



```
a value = 100
a addr  = 65500
p value = 65500
p addr  = 65502
100
a=200, *p=200
a=300, *p=300
```