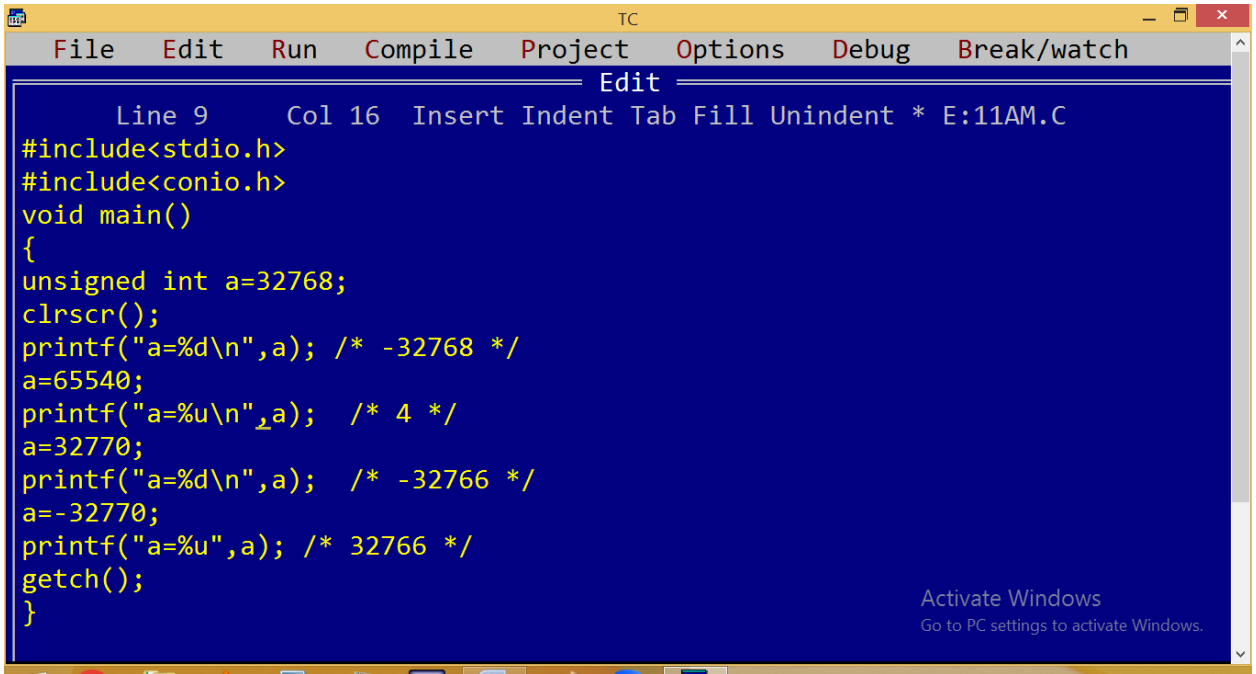
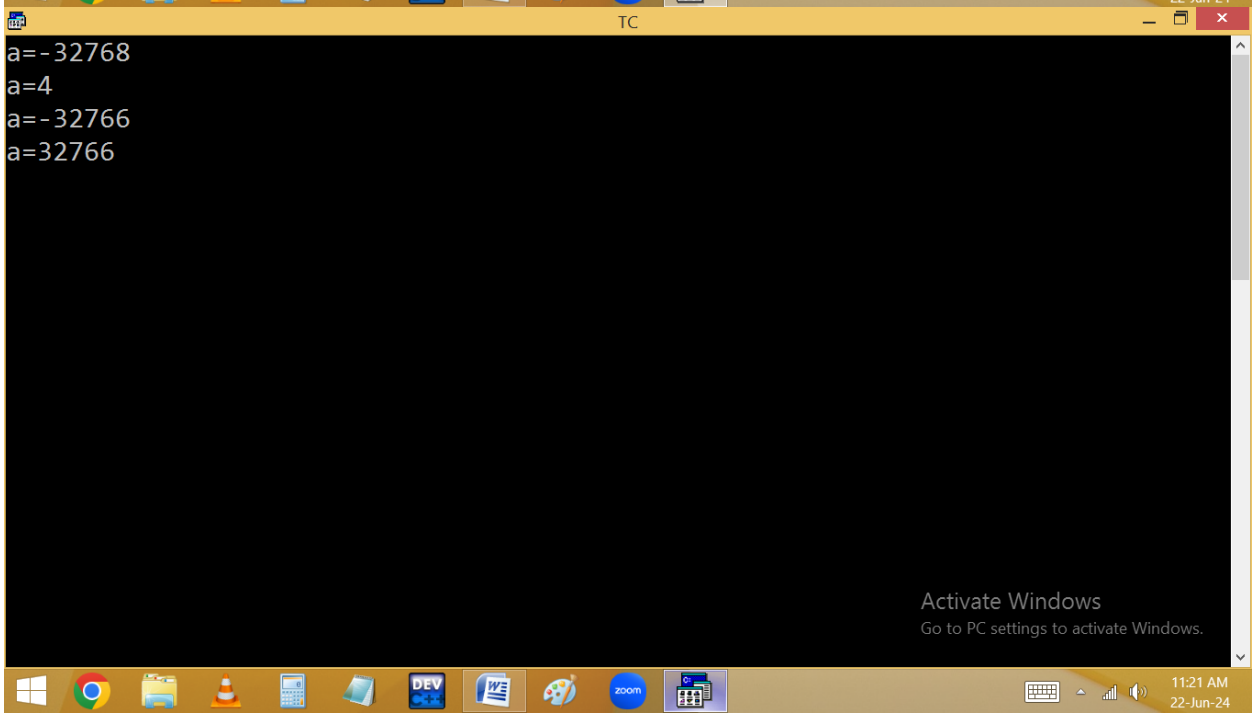
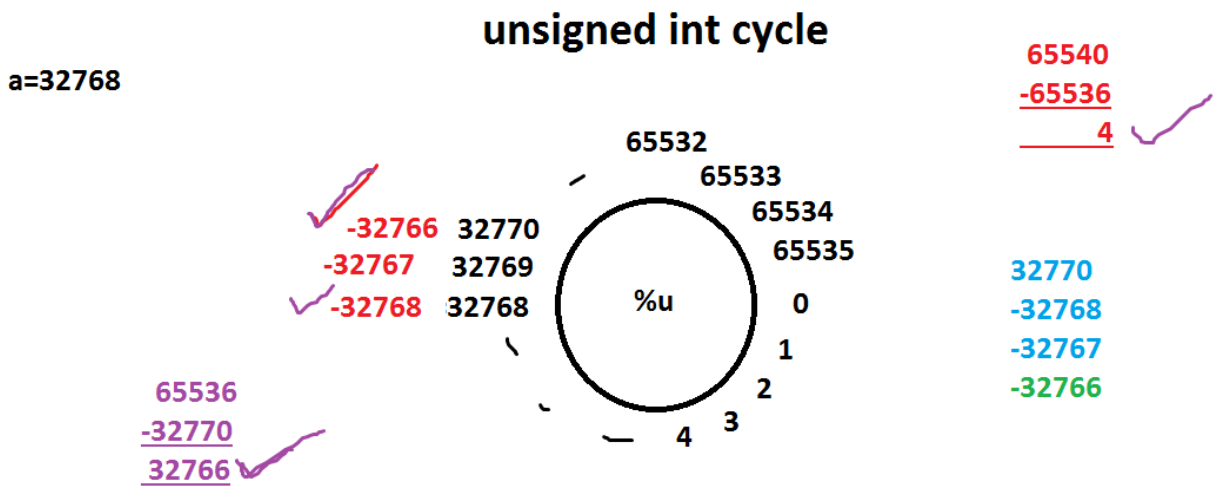

TC
File Edit Run Compile Project Options Debug Break/watch
Edit
Line 9 Col 16 Insert Indent Tab Fill Unindent * E:11AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
unsigned int a=32768;
clrscr();
printf("a=%d\n",a); /* -32768 */
a=65540;
printf("a=%u\n",a); /* 4 */
a=32770;
printf("a=%d\n",a); /* -32766 */
a=-32770;
printf("a=%u",a); /* 32766 */
getch();
}
Activate Windows
Go to PC settings to activate Windows.
11:21 AM
22-Jun-24


TC
a=-32768
a=4
a=-32766
a=32766
Activate Windows
Go to PC settings to activate Windows.
11:21 AM
22-Jun-24

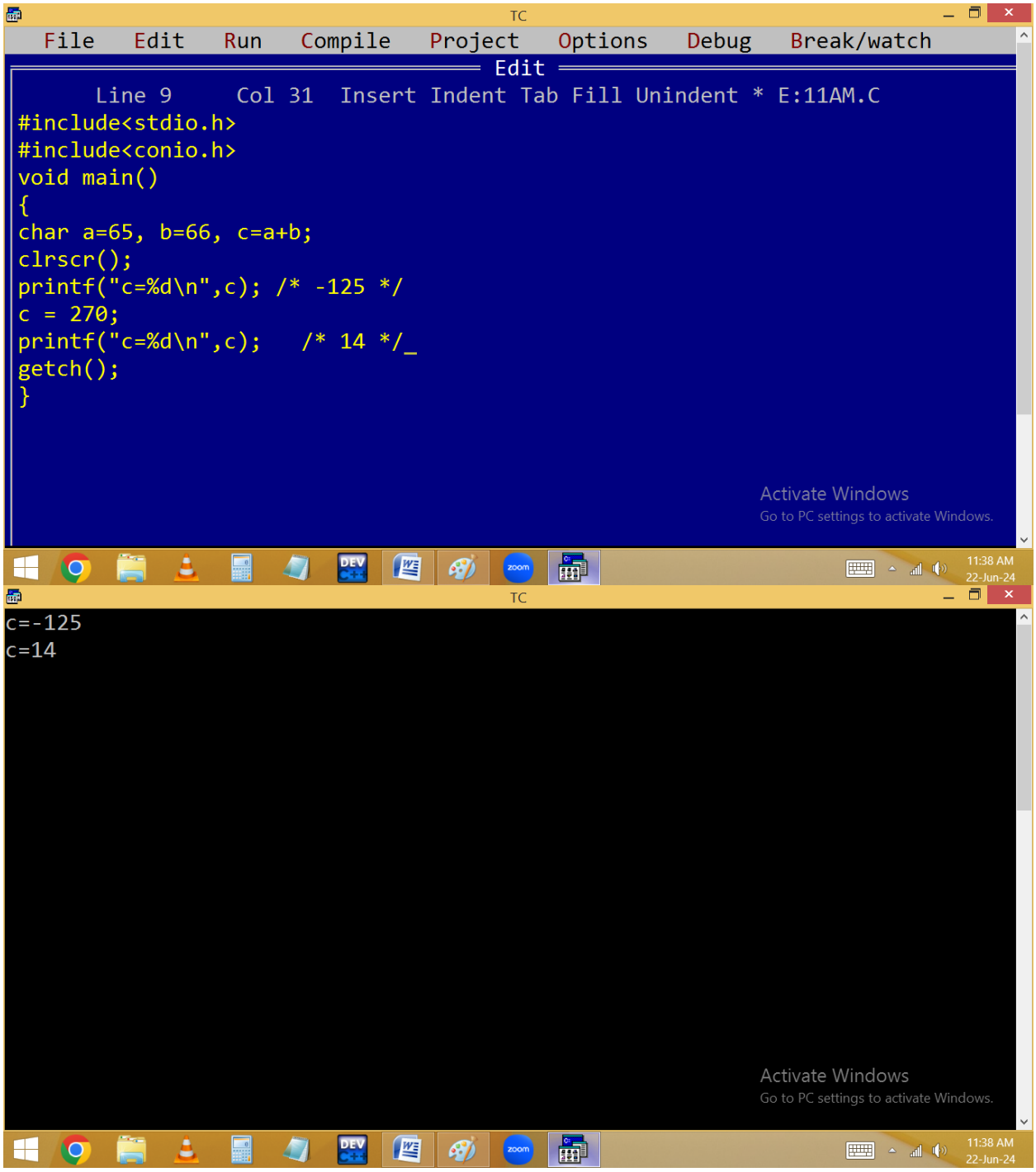


C uses ascii character set ➔ 256 characters

256/2 = 128

Signed char ➔ -128 to +127

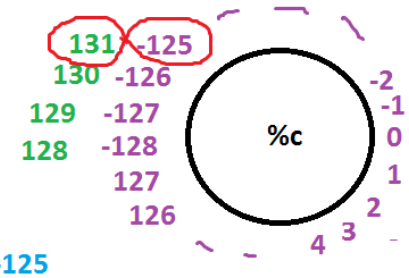
Unsigned char ➔ 0 to 255


TC
File Edit Run Compile Project Options Debug Break/watch
Edit
Line 9 Col 31 Insert Indent Tab Fill Unindent * E:11AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char a=65, b=66, c=a+b;
clrscr();
printf("c=%d\n",c); /* -125 */
c = 270;
printf("c=%d\n",c); /* 14 */_
getch();
}
Activate Windows
Go to PC settings to activate Windows.
11:38 AM
22-Jun-24
TC
c=-125
c=14
Activate Windows
Go to PC settings to activate Windows.
11:38 AM
22-Jun-24

char cycle 256 ascii char

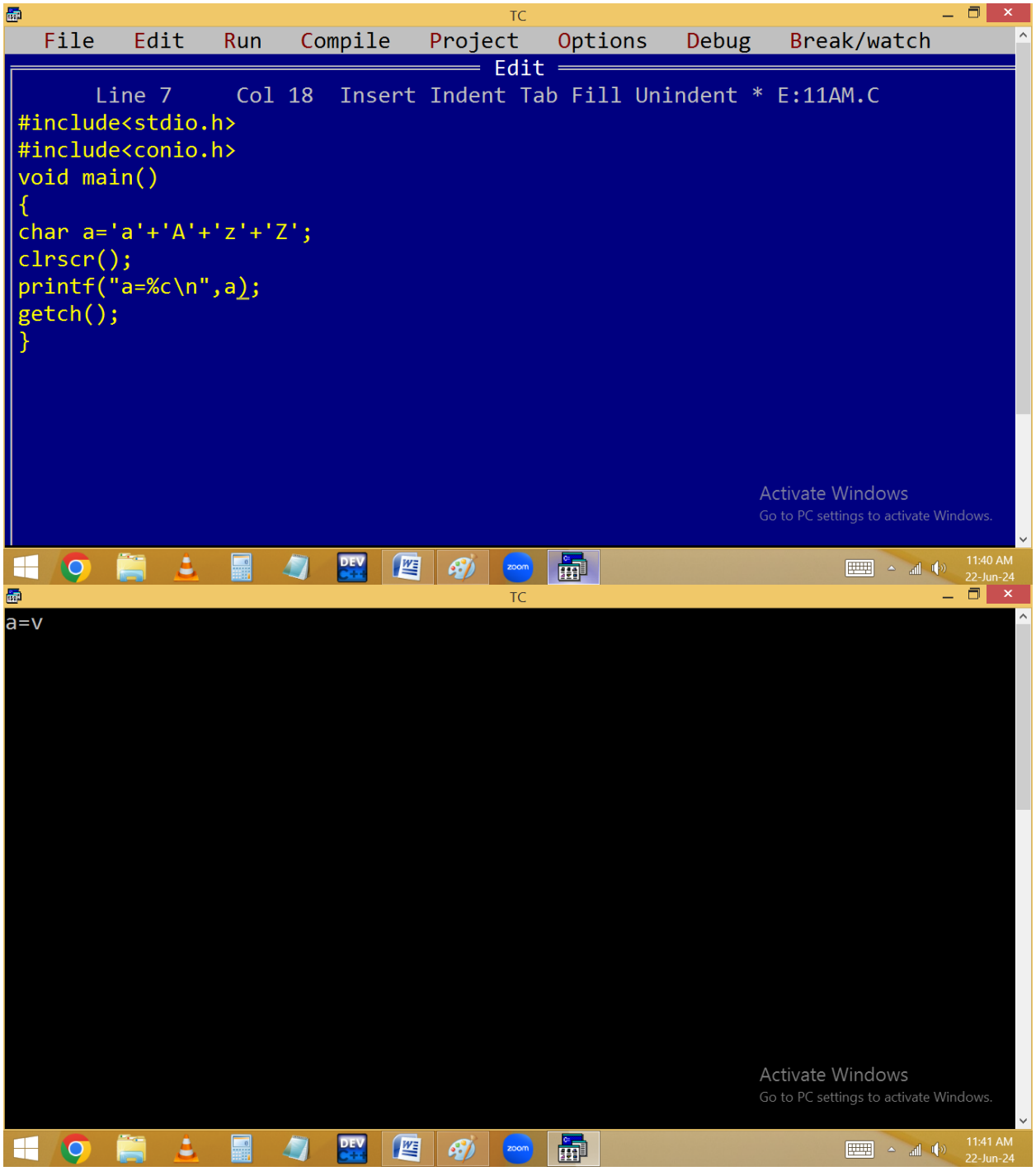
a=65
b=66
c=131

256
-131
-125



131
-128 + 3 = -125
3

270
-256
14


TC
File Edit Run Compile Project Options Debug Break/watch
Edit
Line 7 Col 18 Insert Indent Tab Fill Unindent * E:11AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char a='a'+'A'+'z'+'Z';
clrscr();
printf("a=%c\n",a);
getch();
}
Activate Windows
Go to PC settings to activate Windows.
11:40 AM
22-Jun-24
TC
a=v
Activate Windows
Go to PC settings to activate Windows.
11:41 AM
22-Jun-24

97+65+122+90=374

374
-256
118 = v

VARIABLES

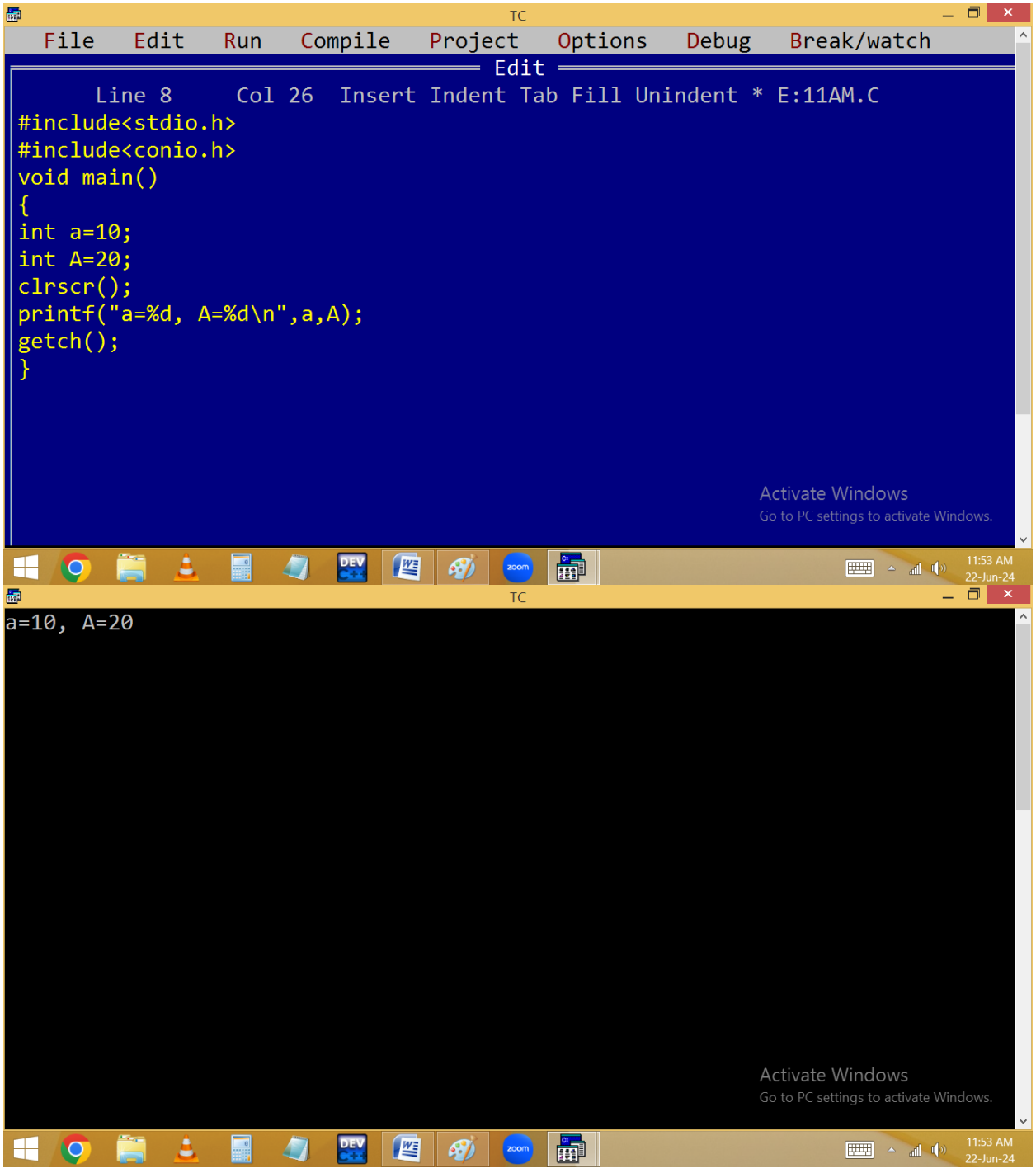
Variable is a container is used to store the values in our programs.

Variable is a named memory location where we can store and manipulate [modify] the values in our programs.

Every variable is going to store in the primary memory i.e. **RAM**, which is temporary. Due to this the variable deleted automatically after the function / program execution.

In c compiler always the variables declared in first line only. In C++ we can declare anywhere.

Variables are case sensitive. i.e. lower and upper are different.



Every variable is having 2 stages.

1. Variable declaration / declared

   Eg: int a;

2. Variable initialization / defined

a=100;

When a variable is initialized then only memory allocated.

Syntax:

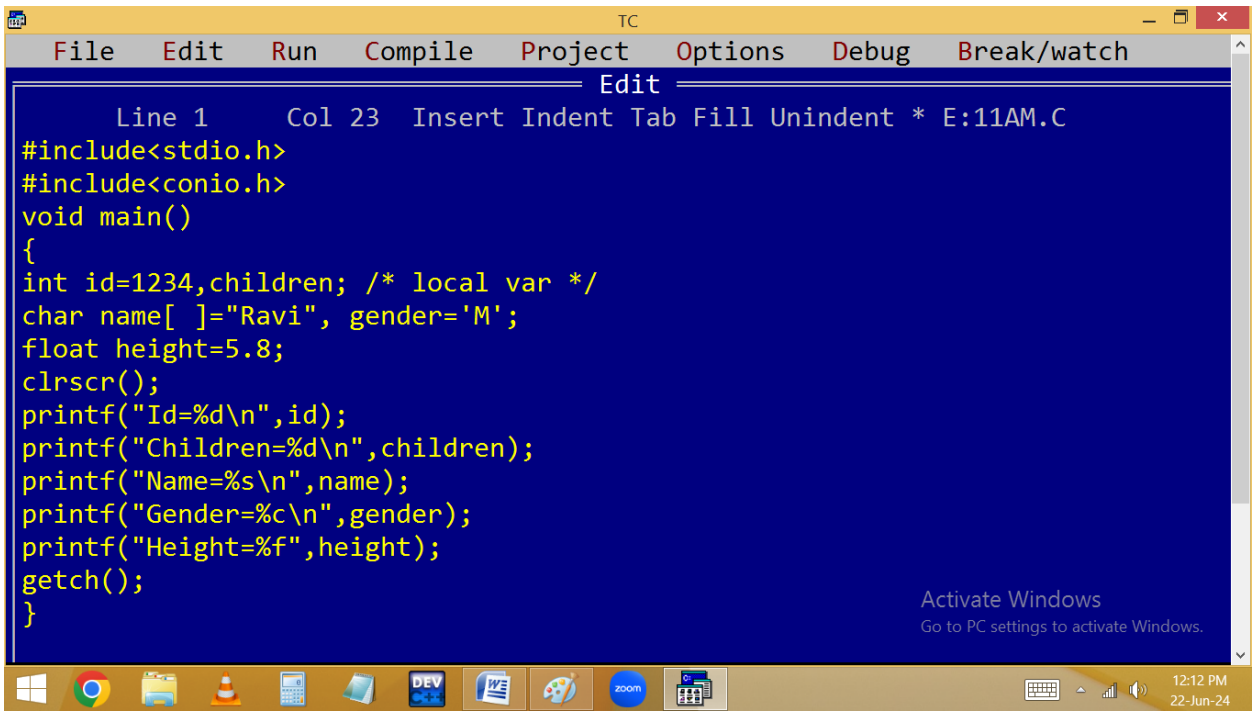
datatype variable[=value], variable[=value],…..**;**

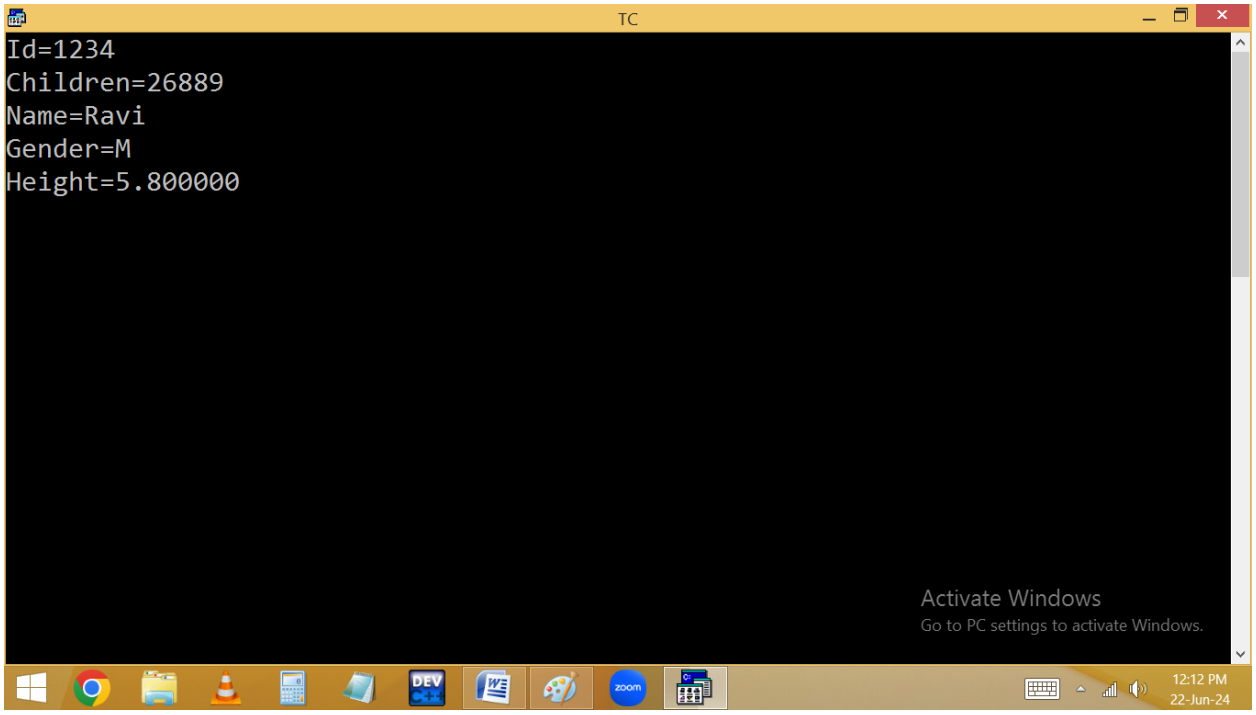
Eg:

int id=1234, children=2;

char name[ ]=”Ravi”, gender=’M’;

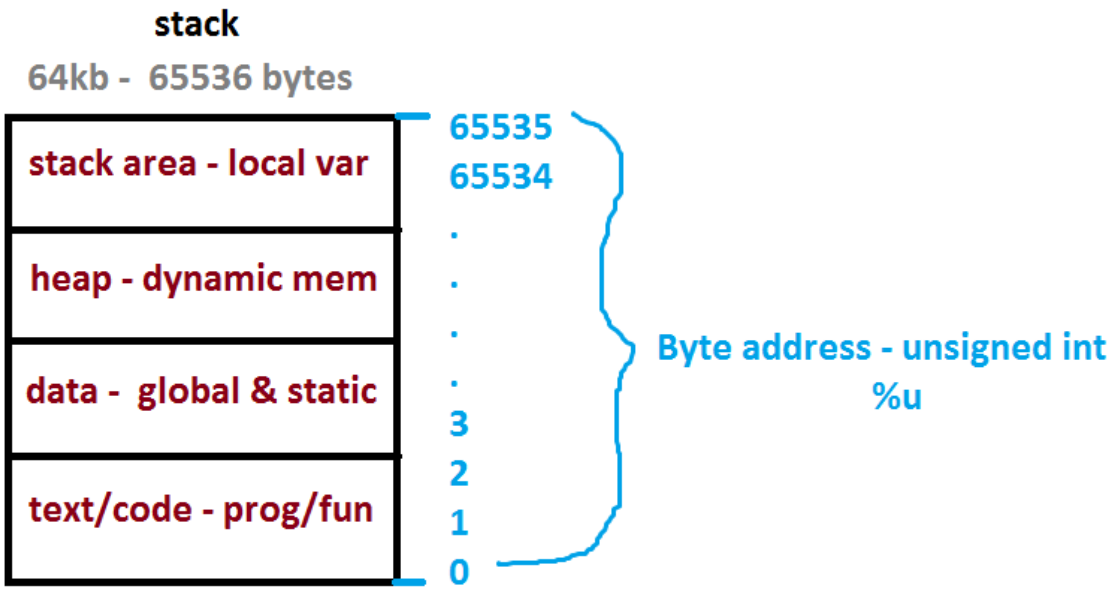
float height = 5.8;

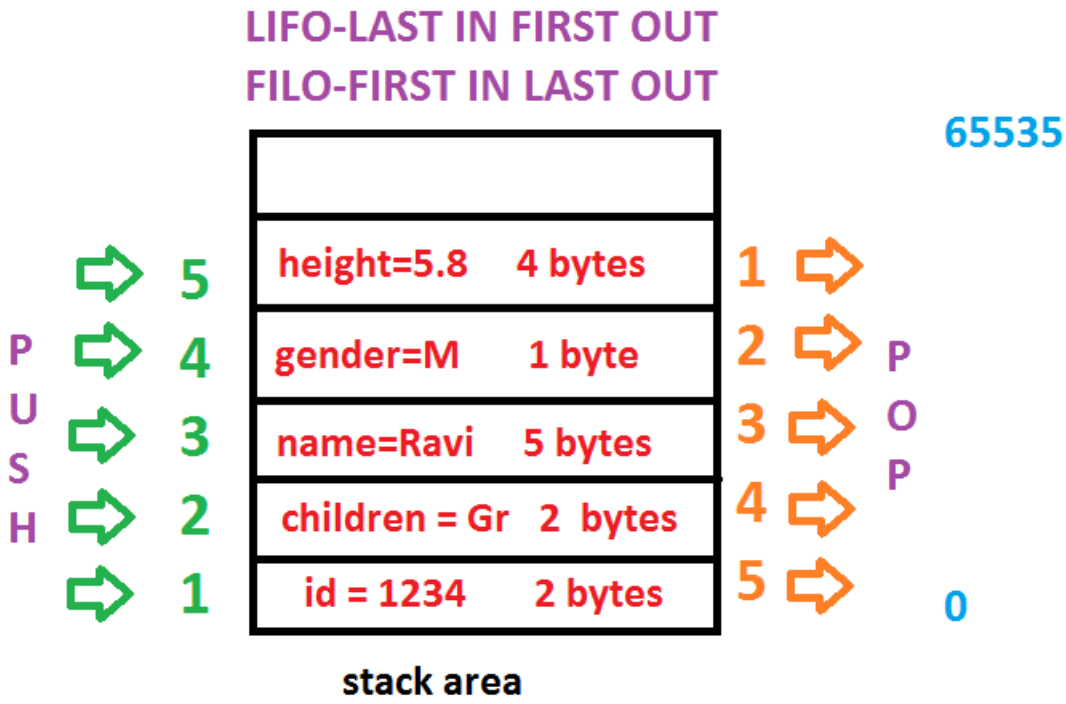




Memory allocation for variables:

4 bits = 1 nibble
2 nibbles / 8 bits = 1 byte
1024 bytes = 1kb
1024kb = 1 mb
1024mb = 1 gb
1024gb = 1 tb



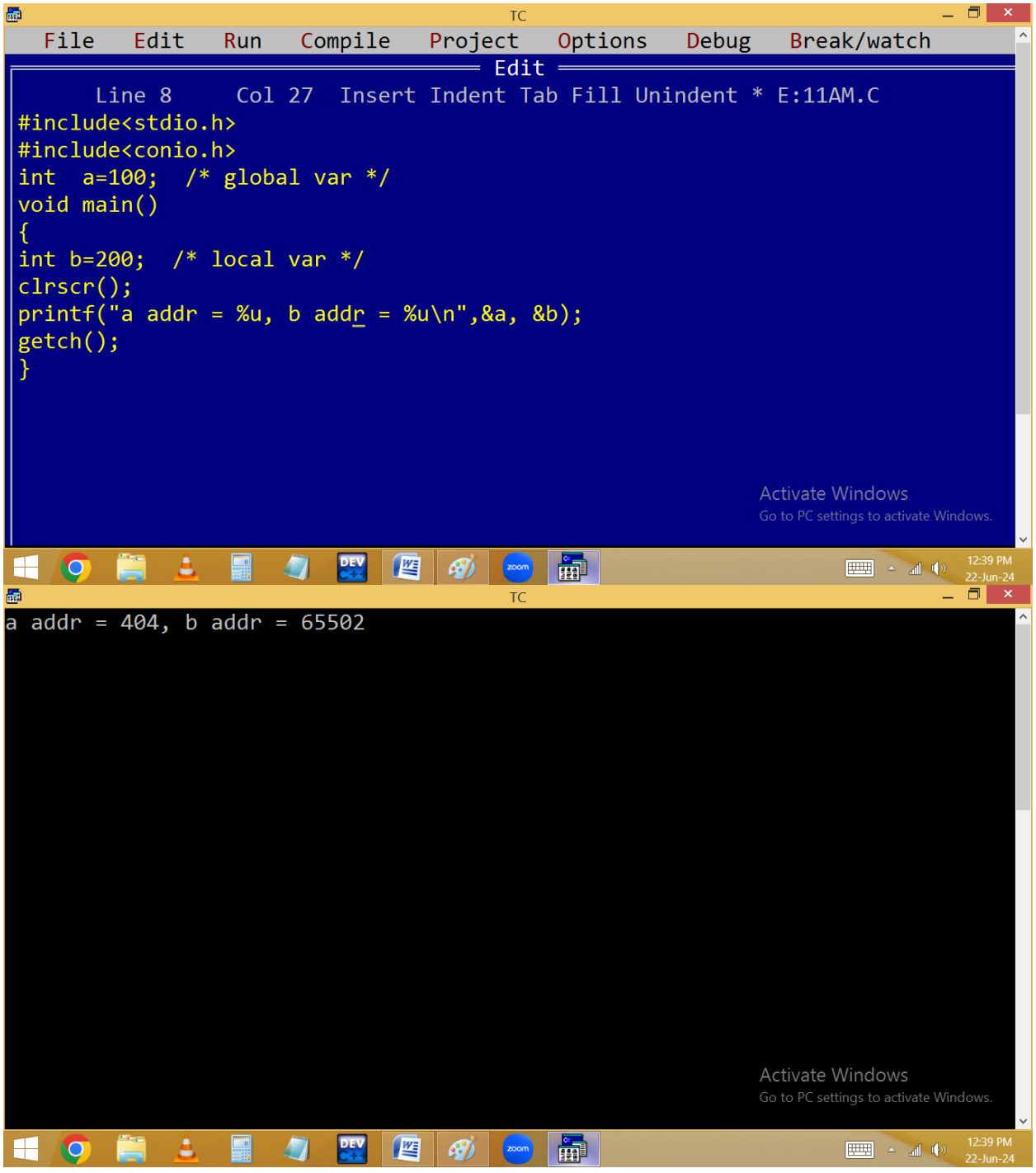


Basically the variables divided into 2 types.

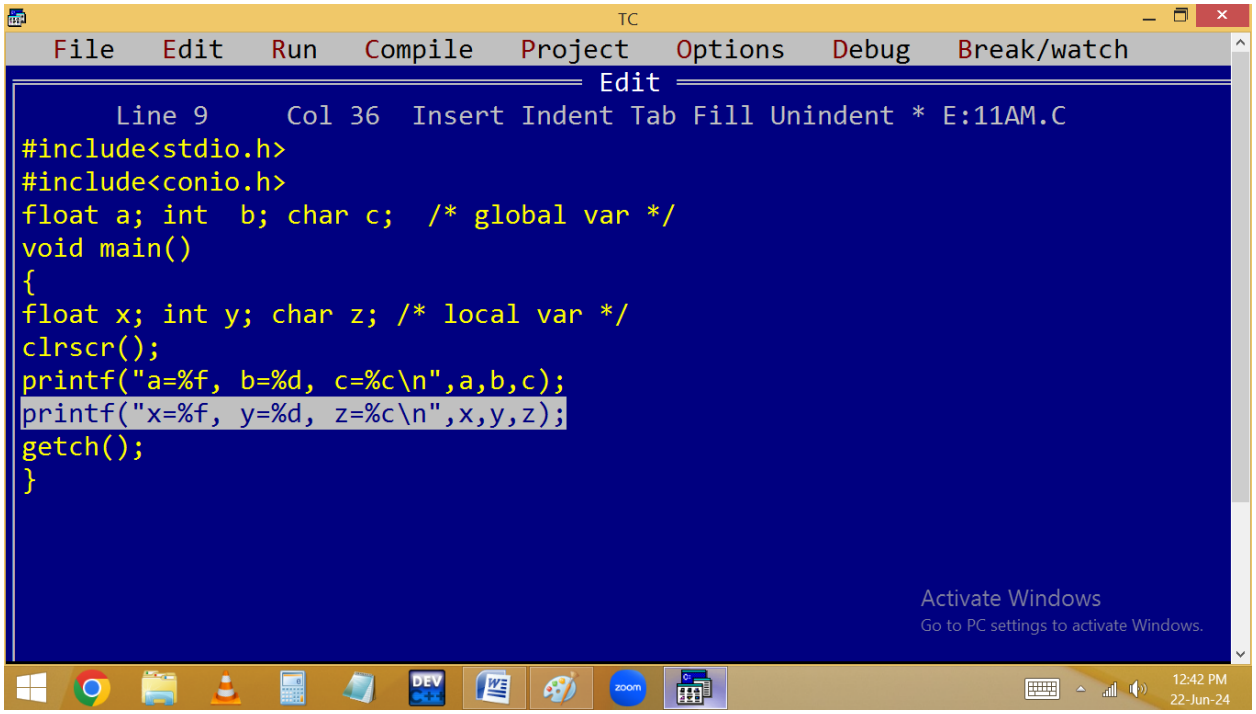
1. Local variables
2. Global variables

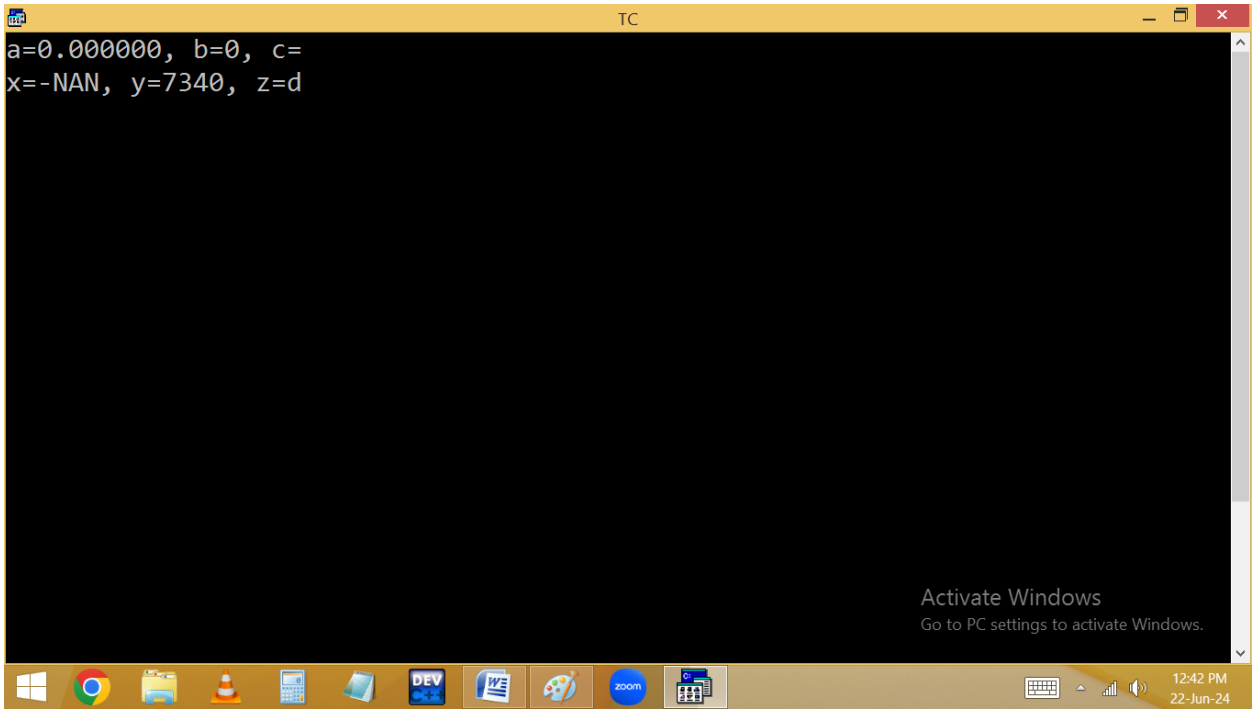
	Local variables	Global variables
Declaration	Within function /{}	Top of the program
Storage area	Stack area	Data area
Initial values [auto values]	Garbage values	Int-0, float-0.000000, char-blank
Scope [availability]	Within fun/{ } only	Anywhere
Life time [active in memory]	Until fun / {} execution	Until main() closed Tot program closed

Finding address of local and global variables[storage area]:



Finding initial values of local and global variables:


TC
File Edit Run Compile Project Options Debug Break/watch
Edit
Line 9 Col 36 Insert Indent Tab Fill Unindent * E:11AM.C
#include<stdio.h>
#include<conio.h>
float a; int b; char c; /* global var */
void main()
{
float x; int y; char z; /* local var */
clrscr();
printf("a=%f, b=%d, c=%c\n",a,b,c);
printf("x=%f, y=%d, z=%c\n",x,y,z);
getch();
}
Activate Windows
Go to PC settings to activate Windows.
12:42 PM
22-Jun-24


TC
a=0.000000, b=0, c=
x=-NAN, y=7340, z=d
Activate Windows
Go to PC settings to activate Windows.
12:42 PM
22-Jun-24