Storage Class in C

What is Storage class in C?

a Storage class represents the visibility and location of a variable. It tells from what part of code we can access a variable. A Storage class in C is used to describe the following things:

- The variable Scop
- The location where the variable will be stored
- The initialized value of a variable
- A lifetime of a variable.

Auto Storage class in C

The variable defined using auto storage class are called as locale variables.
Auto stands for automatic Storage class.

A variable is in auto Storage class by default if it is not explicitely specified

The scope of an auto variable is limited with the particular block only. once the control goes out of the block, the access is destroyed. This means only the block in which the auto variable is declared can access it.

```
#include <stdio.h>
void fun()
{
   auto int a=10;
   ++a;
   Printf("%d", a);
}
int main()
{
   fun();
   fun();
   fun();
   return 0;
}
```

OUTPUT
11
11
11

Static Storage Class in C

The Static variable are used within function/file as local static variables. They can also be used as an global variables.

Static variables have the property of preserving their values even after they are out of their Scope!

Hence Static variables Preserve the value of their last use in their Scope. So we can say that they are initialized only once and exist till the termination of the program. Thus no new memory is allocated because they are not re-declared.

```
#include<stdio.h>
void fun()
{
   static int a=10;
   ++a;
   printf("%d", a);
}
int main()
{
   fun();
   fun();
   fun();
}
```

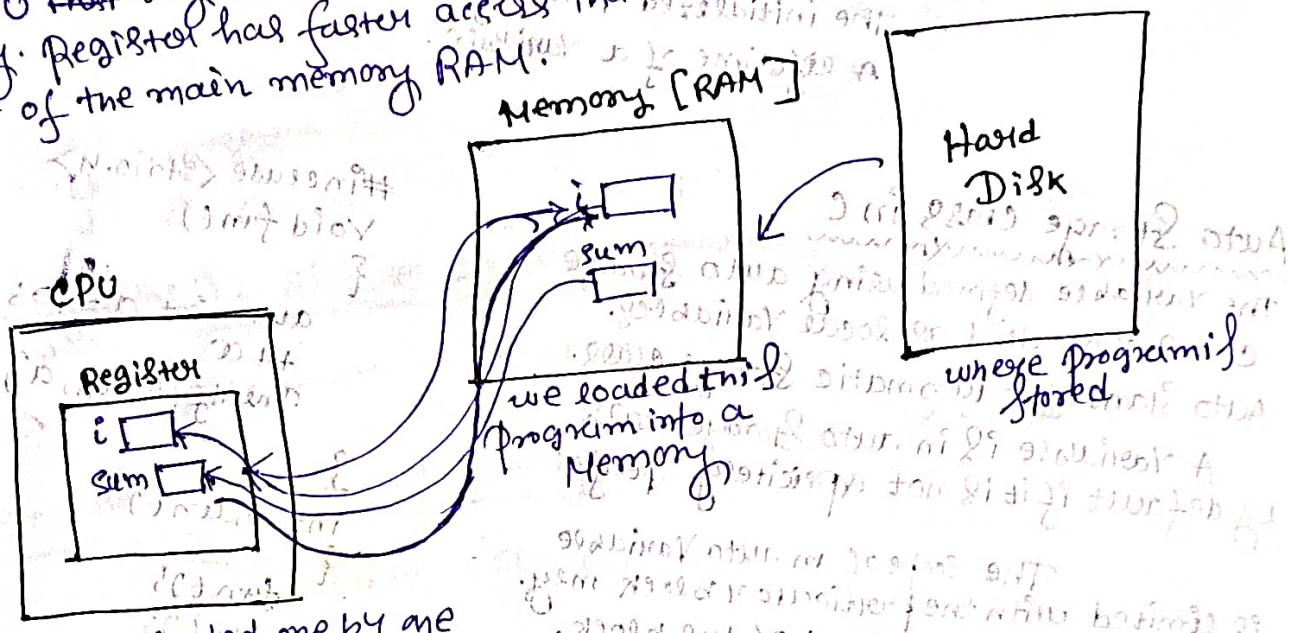
OUTPUT
11
12
13

Register Storage Class in C

The keyword register is used to declare a register storage class. The variables declared using register storage class has lifespan throughout the program.

It is similar to the auto storage class. The variable is limited to the particular block. The only difference if that the variables declared using register storage class are stored inside CPU ~~instt~~ registers instead of a memory. Register has faster access than that of the main memory RAM.

```
#include <stdio.h>
int main()
{
register int i, sum=0;
for(i=0; i<10; i++)
{
  sum = sum+i;
}
Printf("sum=%d", sum);
}
```



all Instruction loaded one by one in this Register and Processing done here.