FUNCTION OVERLOADING

```
void volume ( int );
void volume ( int, int, int );
```

Arg

① No g arg

② type g arg

③ order g arg

```
Select a figure:
1. Cube
2. Cuboid
Enter your choice: 2
Enter l, b, h : 3 4 5
Vol g cub : 60
```

```
void area ( int );
void area ( double );
```

```
void show ( int, char * );
void show ( char *, int );
```

```
int display ( );
float display ( );
```

Function overloading
==========
In C++ the word overloading means having multiple versions of the same entity in the same scope.

Thus the word function overloading means having multiple functions with the same name and within the same scope.

Rules of Function Overloading
==========
When we overload two or more functions, the compiler expects that these functions must differ from each other in terms of their arguments and this difference can be on 3 types

1. Difference in number of arguments
For ex: **void volume(int);**
void volume(int,int,int);

2. Difference in Datatypes of arguments
For ex: **void area(int);**
void area(double);

3. Difference in order of arguments
For ex: **void display(int, char *);**
void display(char *,int);

Special Point
==========
We must remember that when we overload two or more functions then this overloading can only be done on the bases of their arguments and not on the bases of their return types. This means that if two functions within the same scope have same name and same arguments then even if they have different return types still the compiler will give SYNTAX ERROR. So the following two declarations will not even compile:
int show();
float show();
ERROR

```
#include <iostream.h>
#include <conio.h>
void volume(int);
void volume(int,int,int);
int main()
{
  int choice;
  int l,b,h;
  int s;
  cout<<"Select a figure:";
  cout<<endl<<"1.Cube"<<endl<<"2.Cuboid";
  cout<<endl<<"Enter your choice:";
  cin>>choice;
  switch(choice)
  {
     case 1:
          cout<<"Enter side of the cube";
          cin>>s;
          volume(s);
          break;
case 2:
          cout<<"Enter l,b,h of the cuboid
          cin>>l>>b>>h;
          volume(l,b,h);
          break;
default:
          cout<<"Wrong choice";
 }
getch();
return 0;
}
void volume(int s)
{
  cout<<"Vol of cube is "<<s*s*s;
}
void volume(int l,int b,int h)
{
  cout<<"Vol of cuboid is "<<l*b*h;
}
```

O.S.

+ 2

fn call

math.h

```
int abs(int);
long int abs(long int);
float abs(float);
```

```
#include<math.h>

int x = -1;
cout<<abs(x);
```