### 定义：

C++重载，即 同作用域下，同函数名，参数不同（包括类型、个数、顺序）

在同一个作用域内，可以声明几个功能类似的同名函数，但是这些同名函数的形式参数（指参数的个数、类型或者顺序）必须不同。

cout<< 与C中的<< 就是一种重载的运用,原本C中的<<是左移操作符

### 注意事项

* 1. 重载一定要谨慎，不能无的放矢
  2. 重载越多，程序越难看懂
  3. 区分重载与覆盖：
     1. 重载参数不同，覆盖名字、参数相同
  4. 您不能仅通过返回类型的不同来重载函数

C++调用一个函数是可以忽略其返回值的，这种情况下编译器就无法根据返回值类型来确定调用哪一个函数。      所以，重载不能用返回值类型来区别。

* 1. 重载的目的在于方便对不同类型进行同样的处理

### 课后习题作业

/\*\*

\* reload.cpp 函数的重载课后作业

\*

\* @author Darbuly 970073804@qq.com

\* @copytight 2019-2020 DBL

\*/

#include <iostream>

int calc(int num);

int calc(int a,int b);

int calc(int a,int b,int c);

int main()

{

int n[3],cnt=0,res,i=0;

std::cout<<"input:";

do{

std::cin>>n[i++];

cnt++;

}while(std::cin.get()!='\n'||cnt>3);

switch(cnt)

{

case 1:

res = calc(n[0]);

break;

case 2:

res = calc(n[0],n[1]);

break;

case 3:

res = calc(n[0],n[1],n[2]);

break;

}

std::cout<<"\n"<<"res is :"<<res<<std::endl;

return 0;

}

int calc(int num)

{

return num\*num;

}

int calc(int a,int b)

{

return a\*b;

}

int calc(int a,int b,int c)

{

return a+b+c;

}

