CSDN 学院

C++语言基础

迁者 - 贺利坚

http://blog.csdn.net/sxhelijian/

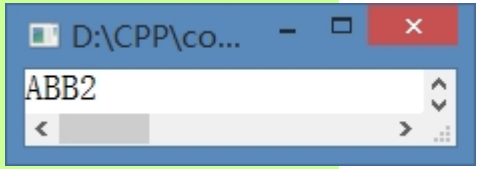
http://edu.csdn.net

本节主题：

函数中的引用
- 引用作为形参
- 引用作为返回值

引用作为形参

```
class Sample
{
   int x;
public:
   Sample(int a): x(a) {cout<<"A";}
   Sample(Sample &a): x(a.x) {cout<<"B";}
   int getX(){return x;}
};
void disp(Sample s){cout<<s.getX();}
int main()
{
   Sample s1(2),s2(s1);
   disp(s1);
   return 0;
}
```




```
class Sample
{
   int x;
public:
   Sample(int a): x(a) {cout<<"A";}
   Sample(Sample &a): x(a.x) {cout<<"B";}
   int getX(){return x;}
};
void disp(Sample &s){cout<<s.getX();}
int main()
{
   Sample s1(2),s2(s1);
   disp(s1);
   return 0;
}
```

函数返回值——简单的返回值说起

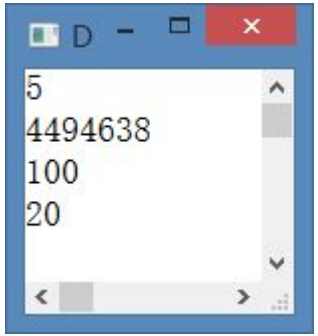
```
#include <iostream>
using namespace std;
int aaa()
{
    int a = 5;
    return a; //值
}

int *bbb()
{
    int b[5]= {0};
    return b;
}
```

```
int *ccc()
{
    static int c = 100;
    return &c;
}

int *ddd()
{
    int *p = new int(20);
    return p;
}
```

```
int main()
{
    int n = aaa();
    int *p1 = bbb();
    int *p2 = ccc();
    int *p3 = ddd();
    int b = 38;
    cout<<n<<endl;
    cout<<*p1<<endl;
    cout<<*p2<<endl;
    cout<<*p3<<endl;
    delete p3;
    return 0;
}
```



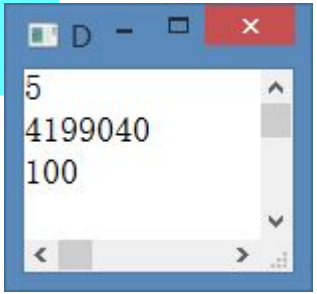
返回值为引用

```
#include <iostream>
using namespace std;
int aaa()
{
    int a = 5;
    return a; //值
}

int &bbb()
{
    int b = 0;
    return b;
}
```

```
int &ccc()
{
    static int c = 100;
    return c; //值
}
```

```
int main()
{
    int n1 = aaa();
    int &n2 = bbb();
    int &n3 = ccc();
    cout<<n1<<endl;
    cout<<n2<<endl;
    cout<<n3<<endl;
    return 0;
}
```

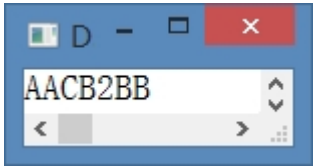


返回值为非引用对象，返回值直接取栈中的结果

```
#include<iostream>
using namespace std;
class Sample
{
   int x;
public:
   Sample(){}
   Sample(int a): x(a) {cout<<"A";}
   Sample(Sample &a): x(a.getX()) {cout<<"D";}
   ~Sample(){cout<<"B";}
   int getX(){return x;}
};
Sample copySample(Sample &a)
{
   Sample b(a.getX());
   cout<<"C";
   return b;
}
void disp(Sample &s){cout<<s.getX();}
```

函数的返回值是类的对象，函数执行返回调用时，不会调用复制构造函数。

编译器自动进行优化。编译器知道这个临时对象是返回的，直接将其放置在需要返回的栈里面，程序返回结果直接取那个栈结果，就省去了一次复制构造。这个是C++的常用优化手段

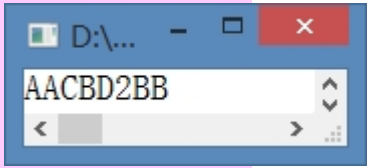



```
int main()
{
   Sample s1(2),s2;
   s2=copySample(s1);
   disp(s2);
   return 0;
}
```

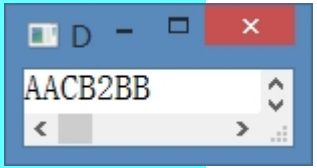
返回值为引用对象时

```
class Sample
{
   int x;
public:
   Sample(){}
   Sample(int a): x(a) {cout<<"A";}
   Sample(Sample &a): x(a.getX()){cout<<"D";}
   ~Sample(){cout<<"B";}
   int getX(){return x;}
};
Sample& copySample(Sample &a)
{
   Sample b(a.getX());
   cout<<"C";
   return b;
}
void disp(Sample &s){cout<<s.getX();}
```

```
int main()
{
   Sample s1(2),s2=copySample(s1);
   disp(s2);
   return 0;
}
```




```
int main()
{
   Sample s1(2),s2;
   s2=copySample(s1);
   disp(s2);
   return 0;
}
```



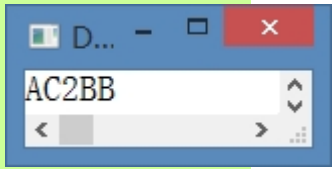
warning: reference to local variable 'b' returned [enabled by default]|

可以这样做！

```
#include<iostream>
using namespace std;
class Sample
{
   int x;
public:
   Sample(){}
   Sample(int a): x(a) {cout<<"A";}
   Sample(Sample &a): x(a.getX()){cout<<"D";}
   ~Sample(){cout<<"B";}
   int getX(){return x;} const
   void setX(int i){x=i;}
};
```

```
Sample& copySample(Sample &a, Sample &b)
{
   b.setX(a.getX());
   cout<<"C";
   return b;
}
void disp(Sample &s){cout<<s.getX();}
```

```
int main()
{
   Sample s1(2),s2;
   s2=copySample(s1,s2);
   disp(s2);
   return 0;
}
```



CSDN 学院

THANKS

本课程由 迁者-贺利坚 提供

CSDN网站：www.csdn.net
企业服务：http://ems.csdn.net/
人才服务：http://job.csdn.net/
CTO俱乐部：http://cto.csdn.net/
高校俱乐部：http://student.csdn.net/
程序员杂志：http://programmer.csdn.net/

CODE平台：https://code.csdn.net/
项目外包：http://www.csto.com/
CSDN博客：http://blog.csdn.net/
CSDN论坛：http://bbs.csdn.net/
CSDN下载：http://download.csdn.net/