验证C++j继承中父类与子类之间构造函数与析构函数的调用顺序：

程序代码如下：

#include <iostream>

using namespace std;

class M

{

public:

int i;

M(){

cout<<"this is M"<<endl;

}

~M(){

cout<<"this is a M"<<endl;

}

protected:

private:

};

class N:public M

{

public:

N(int a){

cout<<"this is N"<<endl;

}

~N(){

cout<<"this is a N"<<endl;

}

protected:

private:

};

int main()

{

N(1);

return 0;

}

运行结果截图：



结论：c++中对象都是先构造成员变量，再调用自身的构造函数；故父类成员变量首先被构造，然后调用父类的构造函数，再构造子类的成员变量，最后调用子类的构造函数。

对象析构时与构造顺序相反，子类先调用自身的析构函数，再调用子类成员变量的析构函数，再调用父类的析构函数，再析构父类的成员变量。