类的模板成员函数不能为虚函数：编译器在处理类的定义时就需要确定这个类的虚函数表的大小，模板类不同的参数实例不同的类型，如果某个模板成员函数为虚函数，就会产生不同的虚函数，这就需要编译器知道程序中所有对该虚成员模板函数的调用，这无法做到。

类模板的构造函数：例程序stl\_test1

// Copyright 2017.刘珅珅

// author：刘珅珅

// 模板构造函数

#include <iostream>

using namespace std;

template <class T>

class MyClass {

public:

// 构造函数

MyClass()

{

cout << "调用默认构造函数" << endl;

}

// 拷贝构造函数

MyClass(const MyClass<T>& x)

{

cout << "调用拷贝构造函数" << endl;

}

// 不同类型的拷贝构造函数

template <class U> MyClass(const MyClass<U>& x)

{

cout << "调用不同参数类型的拷贝构造函数" << endl;

}

};

int main(int argc, char\* argv[])

{

MyClass<double> d1;

MyClass<double> d2(d1);

MyClass<int> i3(d1);

return 0;

}

输出为：

调用默认构造函数

调用拷贝构造函数

调用不同参数类型的拷贝构造函数