

二级C 模板：模板的定义 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/133/2021_2022__E4_BA_8C_E7_BA_A7C___E6_c97_133532.htm

若一个程序的功能是对某种特定的数据类型进行处理，则将所处理的数据类型说明为参数，就可以把这个程序改写为模板。C 程序由类和函数组成，所以 C 的模板也分为类模板和函数模板。

1、函数模板的定义：
说明：可以包含基本数据类型，也可以包含类类型。若是类类型，则须加前缀 class。这样的函数模板定义不是一个实实在在的函数，编译系统不为其产生任何执行代码。该定义只是对函数的描述，表示它每次能单独处理在类型形式参数表中说明的数据类型。当编译系统发现有一个函数调用：FunctionName(实在参数表)。将根据实在参数表中的类型，确认是否匹配函数模板中对应的形式参数表，然后生成一个重载函数。该重载函数的定义体和函数模板的函数定义体相同，而形式参数表的类型则以实在参数表的实际类型为依据。该重载函数称为模板函数。

类模板的一般定义形式：
template class classname
{ // 类声明体 }
template 返回类型

classname::MemberFunctionname1(形式参数表) { // 成员函数定义体 }
template 返回类型 classname::MemberFunctionname2(形式参数表) { // 成员函数定义体 }
template 返回类型

classname::MemberFunctionnamen(形式参数表) { // 成员函数定义体 }

类模板与模板类的区别：类模板是模板的定义，不是一个实在的类，定义中用到通用类型参数。模板类是实在的类定义，是类模板的实例化。类定义中参数被实际

类型所代替。例 3：定义一个单向链表的模板类，它分别实现增加、删除、寻找和打印操作。

```
#include <string>
using namespace std;
template< class T>
class List {
public:
    List() { pFirst=NULL; }
    void Add(T&t) {
        Node* temp=new Node;
        temp->pT=t;
        temp->pNext=pFirst;
        pFirst=temp;
    }
};
```

100Test 下载频道开通，各类考试题目直接下载。
详细请访问 www.100test.com