

二级C 模板：为什么要引入模板 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\\_133530.htm](https://www.100test.com/kao_ti2020/133/2021_2022__E4_BA_8C_E7_BA_A7C___E6_c97_133530.htm) 模板是 C 中相对较新的一个特性。模板使程序员能够快速建立具有类型安全的类库集合和函数集合，它的实现方便了更大规模的软件开发。我们为什么要使用模板呢？首先我们分析这样两种情况：第一种情况是关于函数。考察两个函数，一个交换两个整型数，另一个交换两个浮点数。两个 swap() 的主体行为是一样的，无非一个是处理 int 型的，一个是处理 float 型的。所以可以通过重载函数来实现。两个函数分别如下：  
`void swap(int amp.b) { int temp=a. a=b. b=temp. } swap(floatamp.b) { float temp=a. a=b. b=temp. }` 但是有了模板后，重载就不必要了。第二种情况是关于类。对于 student 类的链表，我们有：  
`class student { //... .. private: float a. } class studentlist { public: studentlist(). void add(studentamp.). ~studentlist(). private: //... }` 该链表类将 student 类对象作为链表结点，进行插入、删除和查找处理。同样如果想处理其他的任何一种类型的对象作为结点的链表，我们必须重新对这链表进行定义。因为所定义的类的行为没有任何变化，只是处理的结点之类型有所不同，所以使得重新定义链表的工作很烦人。但是模板可以解决这个问题。下面就来讲一讲模板的定义和使用。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](http://www.100test.com)