

第九次作业

概念题

1. 简述C++中类属的概念。
2. C++提供了哪两种实现类属函数的机制？简述它们的缺点。
3. 简述C++中参数化多态的概念及作用。

编程题

1. 猜想以下每个调用会匹配到哪个函数模板，以及对应的T是什么数据类型。

```
#include<iostream>
using namespace std;

template <typename T>
void f(T){ cout<<"f(T)"<<endl; }
template <typename T>
void f(const T*){ cout<<"f(const T*)"<< endl; }
template <typename T>
void g(T){ cout<<"g(T)"<<endl; }
template <typename T>
void g(T*){ cout<<"g(T*)"<<endl; }

int main(){
    int a=1;
    int* b=&a;
    const int c=0;
    const int*d=&c;

    f(a);
    f(b);
    f(c);
    f(d);
    g(a);
    g(b);
    g(c);
    g(d);
}
```

2. 链表是很实用的数据结构。为方便实现任意类型数据（<class T>）的单向链表，请创建节点类模板Node和单向链表类模板List。通过List可对链表进行如下操作：可创建一个空链表（没有节点）；可用void add(T dat)在链表末尾添加一个新节点；void display()用来顺序输出所有节点的T类型数据；链表生命周期结束能自动释放节点所占空间。请在main函数中使用链表类模板List建立三条链表，分别添加若干节点并把数据显示出来（数据类型T分别为int，double和string）。

（实现建议：在Node中包含一个T类型的数据和指向下个节点的指针，在List中包含链表的结点总数，以及两个分别指向头结点和尾结点的指针）

3. 设矩阵类模板Matrix可对元素值为任意类型数据（<class T>）的矩阵进行如下操作。

(1) Matrix(int m,int n)可创建一个m行n列的矩阵; Matrix(const Matrix &mat)可用mat拷贝创建一个矩阵; void setMatrix()用来给所有矩阵元素赋值; void display()则用来打印矩阵 (例某4*6的int型矩阵打印结果如图) ; void transport()用矩阵的转置替换原矩阵。请完成矩阵类模板Matrix, 创建两个T为int型的矩阵, 并设计程序使用以上操作。

1	2	3	4	5	6
7	8	9	9	9	9
8	8	8	8	7	7
7	7	6	6	6	6

(2) 类模板Matrix中新增了两种矩阵操作: void square()将矩阵与自身的转置相乘的结果打印出来; Matrix operator+(const Matrix &mat)重载运算符"+", 使得两个同大小和类型的矩阵能直接相加。请创建三个T为double型的同大小矩阵, 并设计程序使用这两种操作, 查看结果是否正确。

(注: 本大题如有动态申请内存, 请添加相应释放内存的操作)