1. **完成函数mySort函数的重载以及input 和output函数的定义，使下列代码能够正确运行。**

**int** main(){

**int** a, b, c;

cout << "-----------input 3 datas for test1------" << endl;

cin >> a >> b >> c; // 例如输入 20 8 15

mySort(a, b, c); // 升序排列

cout << a << ' ' << b << ' ' << c << endl; // 输出8 15 20

cout << "-----------input 3 datas for test2------" << endl;

cin >> a >> b >> c; // 例如输入 30, 80, -40

mySort(&a, &b, &c); // 重载函数 升序排列

cout << a << ' ' << b << ' ' << c << endl; //-40 30 80

**const** **int** n = 6;

**int** arr[n];

cout << "-----------input"<<n<<" datas for test3------" << endl;

input(arr, n); // 输入n个整数

mySort(arr, 3); //数组前3个数据升序排列

output(arr, n); //遍历输出数组

**return** 0;

}

1. 输入一个整数n(大于0，默认用户输入合法，无需程序验证),利用new申请长度为n的int型数组，输入n个整数到该数组，数组元素有正数有负数，数组中连续的一个或多个整数构成一个子数组。求所有子数组中元素之和最大的子数组。如多个子串和相同且都最大，输出最后一组即可）
2. **完成如下时间类的定义，并在主函数中对所有成员函数进行测试。**

**class** Time

{

**int** hour; // 0-23

**int** minute; // 0-59

**int** second; // 0-59

**public**:

**void** setTime(); //输入合法时间

**void** setTime(int, int, int); //设置合法时间

**void** showTime(); // 输出 小时：分钟：秒

int judge(const Time& t); //比较时间，并输出靠后的时间

};