

Lab Assignment 11

Lab Grading Policy: Attendance 40%, Score 60%

In case you have difficulty in finishing the exercises on time, you should upload them by **Thursday noon** with a penalty of 20% on your score. No late submission is permitted after that. We will in general post the reference solutions **by Friday**.

Exercise 1 (30%): Please ask the user to input some integer numbers, then output the sorted numbers and also the occurrences of each number. Do you think storing the numbers in a `vector<int>` is a good idea?

You will need to use:

```
cin.clear();  
cin.ignore();
```

to recover the `cin` stream so that you can input again, because the 'a' has made the `cin` in an error state. Note that control + D will terminate the input stream on Linux/Mac, and you can no longer use `cin` anymore, without using another thread....

Below is a sample output:

```
Please input some integers, and end with non-integer input: 1 4 3 2 1 2 a  
1 occurs 2 times  
2 occurs 2 times  
3 occurs 1 time  
4 occurs 1 time  
What number do you want to remove? 2  
After erasing 2:  
1 occurs 2 times  
3 occurs 1 time  
4 occurs 1 time
```

Exercise 2 (30%): Write a program to sort some Person objects.

```
class Person {  
    private:  
        string fn;    // first name  
        string ln;    // last name  
    public:  
        Person() {}  
        Person(const string& f, const string& n): fn(f), ln(n) {}  
        string firstname() const;  
        string lastname() const;  
        // ...  
};  
  
inline string Person::firstname() const {
```

```
        return fn;
    }

    inline string Person::lastname() const {
        return ln;
    }

    ostream& operator<< (ostream& s, const Person& p){
        s << "[" << p.firstname() << " " << p.lastname() << "];"
        return s;
    }
}
```

The following is the main you can use:

```
int main(){
    // create some persons
    Person p1("nicolai","josuttis");
    Person p2("ulli","josuttis");
    Person p3("anica","josuttis");
    Person p4("lucas","josuttis");
    Person p5("lucas","otto");
    Person p6("lucas","arm");
    Person p7("anica","holle");

    // insert person into collection coll
    map<Person, int> coll;
    coll[p1]++;
    coll[p2]++;
    coll[p3]++;
    coll[p4]++;
    coll[p5]++;
    coll[p6]++;
    coll[p7]++;

    // print elements
    cout << "map includes:" << endl;
    map<Person, int>::iterator pos;
    for (pos = coll.begin(); pos != coll.end(); ++pos) {
        cout << pos->first << ", " << pos->second << endl;
    }
}
```

However, the map container needs to know how to compare between objects of `Person`. Please provide the definition of the `<` operator for `Person`.

```
map includes:  
[lucas arm], 1  
[anica holle], 1  
[anica josuttis], 1  
[lucas josuttis], 1  
[nicolai josuttis], 1  
[ulli josuttis], 1  
[lucas otto], 1
```