CS205 C/ C++ Programming - Lab Assignment 6

Name: 唐千栋(Qiandong Tang)

SID: 11612730

Part 1 - Analysis

The requirement is to design a class named Box whose dimensions are integers and private to the class. The dimensions are labelled: length l, breadth b, and height h.

Apart from the above, the class should have 4 functions:

- int getLength() - Return box's length
- int getBreadth() - Return box's breadth
- int getHeight() - Return box's height
- long long CalculateVolume() - Return the volume of the box

We need to declare three private member variables in class Box, `_length`, `_breadth`, `_height`. `getLength()`, `getBreadth()`, `getHeight()` simply return the box's length, breadth, height. Since the range of each dimension is [0, 100000], the volume of the box can be greater than `INT_MAX`, then we have to multiply each dimension using long long.

Part 2 - Code

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

class Box {
  private:
  int _length;
  int _breadth;
  int _height;
  bool in_range(int x) {return (0 <= x && x <= 100000) ? true : false;}
  public:
  Box() {_length = 0, _breadth = 0, _height = 0;}
  Box(const Box &box) {
    _length = box._length;
    _breadth = box._breadth;
    _height = box._height;
  }
  Box(int length, int breadth, int height) {
    if (in_range(length)) _length = length;
    else cerr << "Input length out of range" << endl;
    if (in_range(breadth)) _breadth = breadth;
    else cerr << "Input breadth out of range" << endl;
```

```
    if (in_range(height)) _height = height;
    else cerr << "Input height out of range" << endl;
  }
  int getLength() const {return _length;}
  int getBreadth() const {return _breadth;}
  int getHeight() const {return _height;}
  long long calculateVolume() {
    return 1LL * _length * _breadth * _height;
  }
  bool operator<(const Box &rhs) const {
    if (_length < rhs._length) return true;
    if (_breadth < rhs._breadth && _length == rhs._length) return true;
    if (_height < rhs._height && _breadth == rhs._breadth && _length ==
rhs._length) return true;
    return false;
  }
  friend ostream &operator<< (ostream &out, const Box &box) {
    out << box._length << " " << box._breadth << " " << box._height;
    return out;
  }
};

```

Part 3 - Result & Verification

Test case:

```
#include <cstdio>
#include <iostream>
#include "Box.h"
using namespace std;

int main() {
  /*
  Expected output:
  b1: 0 0 0
  b2: 1 2 3
  b3: 1 2 4
  b4: 1 2 4
  b3's length = 1
  b3's breath = 2
  b3's height = 4
  b2 is smaller than b3
  Input length out of range
  b6's volume = 6000000000000
  */
  Box b1;
  cout << "b1: " << b1 << endl;
  Box b2(1, 2, 3);
```

```
  cout << "b2: " << b2 << endl;
  Box b3(1, 2, 4);
  cout << "b3: " << b3 << endl;
  Box b4 = b3;
  cout << "b4: " << b4 << endl;
  cout << "b3's length = " << b3.getLength() << endl;
  cout << "b3's breath = " << b3.getBreadth() << endl;
  cout << "b3's height = " << b3.getHeight() << endl;
  if (b2 < b3) cout << "b2 is smaller than b3" << endl;
  else cout << "b2 is not smaller than b3" << endl;
  Box b5(1000000, 1000, 100000);
  Box b6(10000, 20000, 30000);
  cout << "b6's volume = " << b6.calculateVolume() << endl;
  return 0;
}

```

Part 4 - Difficulties & Solutions

1. How to overload operator `<<`

```
friend ostream &operator<< (ostream &out, const Box &box);
```