CS205 C/C++ Programming Lab Assignment 6

Name: 钟兆玮 (Zhaowei Zhong)
SID: 11611722

Part 1 - Analysis

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.
The default constructor of the class should initialize l, b, and h to 0
The parameterized constructor Box (int length, int breadth, int height) should initialize Box's l, b and h to length, breadth and height.
The copy constructor Box (const Box& b) should set l, b and h to B's l, b and h, respectively.
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

Overload the operator < for the class Box. Box A < Box B if

1. A.l < B.l
2. A.b < B.b and Al. == B.l
3. A.h < B.h and A.b == B.b and == A.l == B.l

Overload operator << for the class Box().
If B is an object of class Box:
cout << B should print B.l, B.b and B.h on a single line separated by spaces.

Constraints:

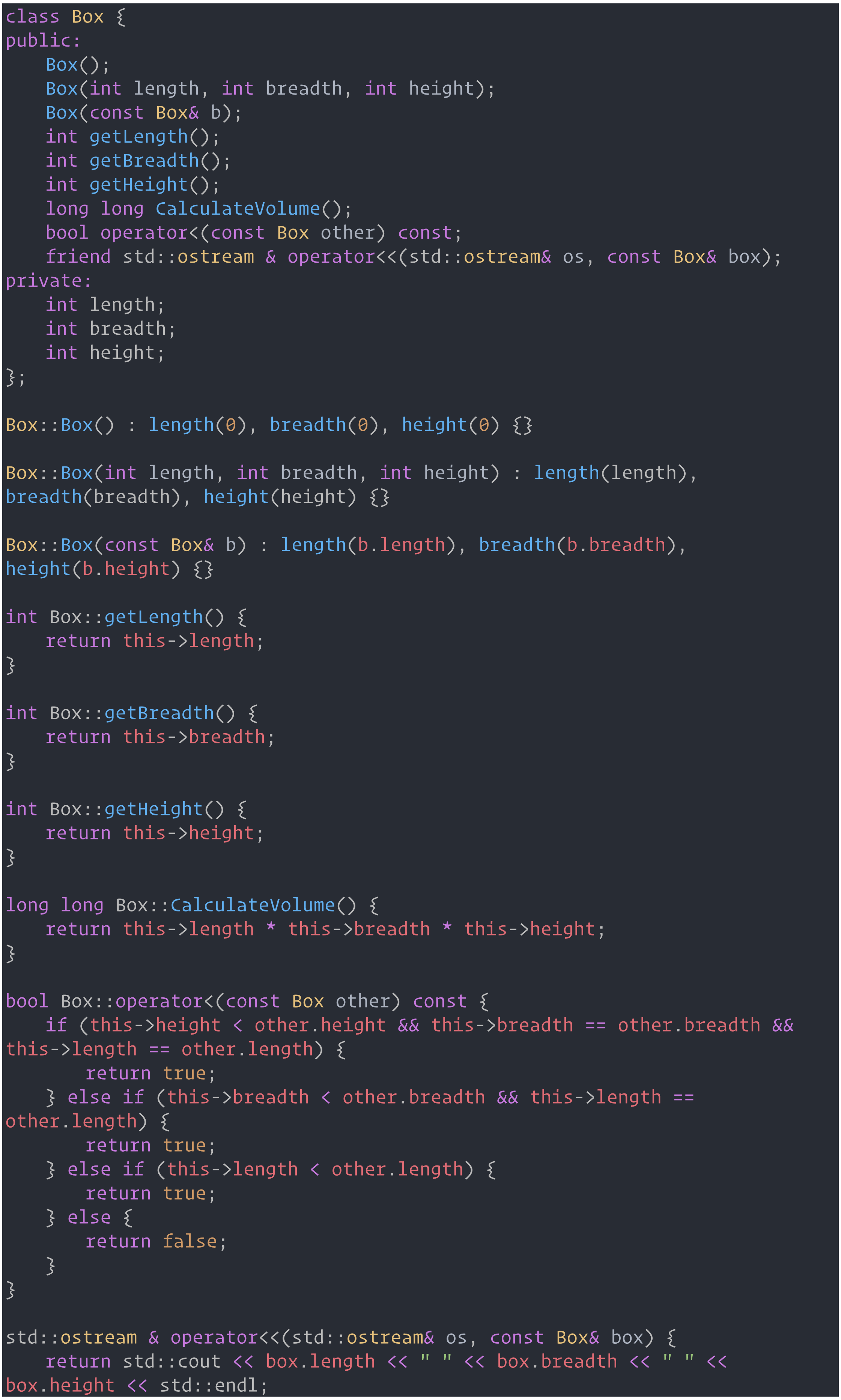
1. l, b and h are in the range of [0, 100000].

Two boxes being compared using the operator will not have all three dimensions equal.

2. Please implement your Box class in box.hpp. You only need to submit box.hpp and your report in this assignment.

Part 2 - Code

```
#ifndef BOX_HPP
#define BOX_HPP
#include <iostream>
```



```
class Box {
public:
    Box();
    Box(int length, int breadth, int height);
    Box(const Box& b);
    int getLength();
    int getBreadth();
    int getHeight();
    long long CalculateVolume();
    bool operator<(const Box other) const;
    friend std::ostream & operator<<(std::ostream& os, const Box& box);
private:
    int length;
    int breadth;
    int height;
};

Box::Box() : length(0), breadth(0), height(0) {}

Box::Box(int length, int breadth, int height) : length(length),
breadth(breadth), height(height) {}

Box::Box(const Box& b) : length(b.length), breadth(b.breadth),
height(b.height) {}

int Box::getLength() {
    return this->length;
}

int Box::getBreadth() {
    return this->breadth;
}

int Box::getHeight() {
    return this->height;
}

long long Box::CalculateVolume() {
    return this->length * this->breadth * this->height;
}

bool Box::operator<(const Box other) const {
    if (this->height < other.height && this->breadth == other.breadth &&
this->length == other.length) {
        return true;
    } else if (this->breadth < other.breadth && this->length ==
other.length) {
        return true;
    } else if (this->length < other.length) {
        return true;
    } else {
        return false;
    }
}

std::ostream & operator<<(std::ostream& os, const Box& box) {
    return std::cout << box.length << " " << box.breadth << " " <<
box.height << std::endl;
}

#endif // BOX_HPP
```

Part 3 - Result & Verification

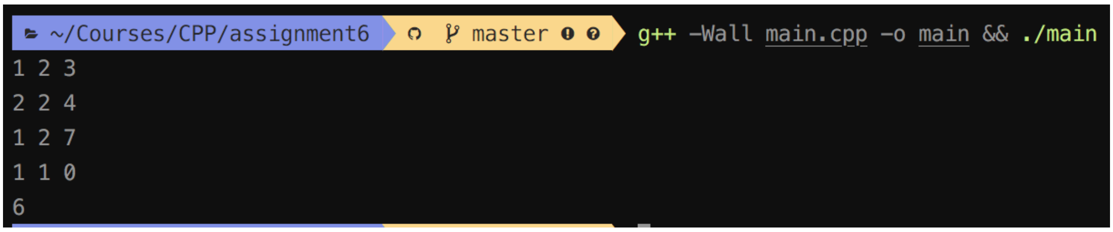
Test Case: main.cpp

```
#include "Box.hpp"
#include <iostream>

using namespace std;

int main() {
    Box box1 = Box(1,2,3);
    Box box2 = Box(2,2,4);
    Box box3 = Box(1,2,7);
    bool comp1 = box1 < box2;
    bool comp2 = box1 < box3;
    bool comp3 = box2 < box3;
    cout << box1 << box2 << box3;
    cout << comp1 << " " << comp2 << " " << comp3 << endl;
    cout << box1.CalculateVolume() << endl;
    return 0;
}
```

Result:



```
~/Courses/CPP/assignment6   master  g++ -Wall main.cpp -o main && ./main
1 2 3
2 2 4
1 2 7
1 1 0
6
```

Part 4 - Difficulties & Solutions

No difficulties.