

CSCI 136 C++ programming

Lab exercise 10

Saad Mneimneh
Computer Science
Hunter College of CUNY

Assume the existence of class Rat with operators + and << properly overloaded.
We would like to create a collection of Rats as follows:

```
class Collection {
private:
    Rat c[100]; //later we will make it dynamic
    int s; //size of collection

public:
    Collection( ... , int s) {
        this->s=s;
        //set c[0]...c[s-1] to the first parameter
    }

    set(int pos, ... ) {
        //set c[pos] to the provided parameter
    }

    int size() {
        return s;
    }

    Rat average() {
        //compute the average of the first s Rats and return it
    }
};
```

(a) Choose how you want to pass the missing parameters, e.g. by value, by reference, const, ...

(b) Use const whenever it is appropriate, i.e. modify the code above by adding const as much as you can.

The class can be used as follows (an example):

```
int main() {
    Collection col=Collection(Rat(),10); //10 default Rats
    col.set(5, Rat(2,3)); //set the fifth element to 2/3

    . . .

    cout<<col.average()<<'\n'; //output the average of the collection
}
```

The important feature of this class is that it contains objects of another class as members (not just basic types). Later, we will modify this class to use dynamic memory allocation (and get rid of the 100 limit on the size), and to be a collection of anything (not just Rats).