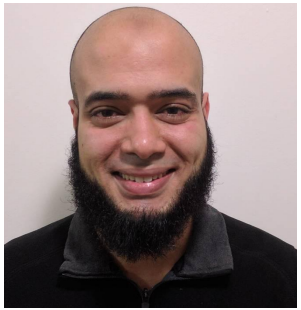
C++ Programming

Operator Overloading

Homework 2

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



Homework 1: Operator &&

```
4  class Boolean {
5  private:
6      bool is_true;
7
8  public:
9      Boolean(bool is_true) : is_true(is_true) {  }
10     bool operator &&(const Boolean &other) const {
11         return this->is_true && other.is_true;
12     }
13 };
14
15 bool T() {  cout << "T\n";  return true;    }
16 bool F() {  cout << "F\n";  return false;   }
17
18 Boolean TC() {  cout << "TC\n"; return Boolean(true);   }
19 Boolean FC() {  cout << "FC\n"; return Boolean(false);  }
20
21 int main() {
22     F() && T();
23     FC() && TC();
```

- Guess the output
- Run the program
- Compare and Guess
- Give a tip

Homework 2: Our Map

```
4  class MyMap {
5  private:
6      vector<string> vec_strs;
7      vector<int> vec_ints;
```

```
52 int main() {
53     MyMap map;
54
55     map["mostafa"] = 20;
56     map["mostafa"] = 40;
57     map["sayed"] = 20;
58     map["ali"] = 20;
59
60     cout << map["mostafa"] << "\n"; // 40
61
62     vector<string> v = map[20];
63     for (auto s : v)
64         cout << s << "\n";  // ali sayed
65
66     map.Reset_iterator();
67     while (map.HasNext()) {
68         auto p = map.GetNext();
69         cout << p.first << " " << p.second << "\n";
70     }
71     map.Clear();
72
```

- Ever wondered from map usage?!
- We want implement our own map <string, int>. Internal data structures is and usage as on right
- **[string]** return corresponding string or 0 if not exist
- **[int]** return sorted vector of strings that has this integer value
- HasNext/GetNext to iterate and print
- Clear: just remove all stored entries

Homework 3: Double

```
35 int main() {
36     double d1 = 1 + 3.0 / 7.0 -1;
37     double d2 = 3.0 / 7.0;
38
39     // 0.428571 0.428571 0
40     // If gave true, play with similar examples
41     cout<<d1<<" "<<d2<<" "<<(d1 == d2)<<"\n";
42
43     Double cd1(d1);
44     Double cd2(d2);
45
46     cout<<(cd1 == cd2)<<"\n";   // true
47
48     map<Double, string> map;
49     map[cd1] = 10;
50     map[cd2] = 20;
51
52     cout<<map.size();   // 1
53
```

- Have u ever failed to compare doubles directly?
  - This is due to double representation
- More worse: Map<double, >
  - Won't work
- Never compare doubles directly
  - Either function to compare: EPS
  - Or Double class
- Create double class that supports operators < > <= >= ==

Homework 4: Chessboard

```
4  class ChessBoard {
5  private:
6      int rows;
7      int colums;
8
9  public:
10     ChessBoard(int rows, int colums) :
11             rows(rows), colums(colums) {
12     }
13 };
14
15 int main() {
16     ChessBoard board(2, 3); // 2x3 grid
17
18     board(0, 0) = 10;
19     board[0][1] = 20;
20
21     cout << board[0][0] << " " << board(0, 1);
22
23     return 0;
24 }
25
```

- We wanna build chessboard class
  - 2d board of integers
- Change this class to allow these operators
  - 2 ways of set and get

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."