Arrays of classes

Here is a very simple program to illustrate an array with a class as the base type. The program keeps information about wind speed and direction for a week. We declare a class `WindInfo`. (Note that we did not use separate compilation here. Both the class definition and the implementation are part of the program.)

Class `WindInfo` has two data members, velocity and direction. We have four member functions, two accessors `getVelocity()` and `getDirection()` to return the values of the member variables, and two mutators `setVelocity()` and `setDirection()` to change the values of the member variables. We need these member functions because we cannot access the values of the member variables directly in a program.

The program stores the values of the wind speed (velocity) and direction for each day in an array `weekWind`. (In the example I only do it for a weekend.) Look at the green highlighted part to see how we use the mutators and accessors to save the values for each day and to retrieve it to display it.

```
#include <iostream>
#include <cstdlib>
 using namespace std;
 //class declaration
 class WindInfo
 {
    public:
        WindInfo();//constructor
        WindInfo(double v, char d); //overloaded constructor
        double getVelocity(); //accessor for velocity
        char getDirection(); //accesor for direction
        void setVelocity(double v); // mutatr for velocity
        void setDirection(char d); //mutator for direction
    private:
        double velocity;//in km per hour
        char direction; //'N', 'S', E', 'W', or '0' if velocity = 0
};

//class implementation
WindInfo::WindInfo()
{
    velocity = 0;
    direction = '0';
}

 WindInfo::WindInfo(double v, char c)
{
    velocity = v;
    direction = 'c';
}

double WindInfo::getVelocity()
{
    return velocity;
}

char WindInfo::getDirection()
```

```
{
    return direction;
}

void WindInfo::setVelocity(double v)
{
    velocity = v;
}

void WindInfo::setDirection(char d)
{
    direction = d;
}

//main program

int main()
{
    WindInfo weekWind[7];
    double vel;
    char dir;

    cout << "Enter windspeed (velocity) and direction for weekend:"
         << endl;
    for (int i = 0; i < 2; i++)
    {
        cout << "Enter velocity for day " << i+1 <<" :";
        cin >> vel;
        weekWind[i].setVelocity(vel);
        cout << "Enter direction for day " << i+1 <<" :";
        cin >> dir;
        weekWind[i].setDirection(dir);
    }

    cout << endl << "Display windspeed (velocity) and direction for"
         << " weekend:" << endl;
    for (int i = 0; i < 2; i++)
    {
        cout << "Windspeed for day " << i+1 <<" :"
             << weekWind[i].getVelocity() << endl;
        cout << "Direction for day " << i+1 <<" :"
             << weekWind[i].getDirection() << endl;

    }

    return 0;

}
```

Output

```
Enter windspeed (velocity) and direction for weekend:
Enter velocity for day 1 :12
Enter direction for day 1 :W
Enter velocity for day 2 :11
```

```
Enter direction for day 2 :E

Display windspeed (velocity) and direction for weekend:
Windspeed for day 1 :12
Direction for day 1 :W
Windspeed for day 2 :11
Direction for day 2 :E

Process returned 0 (0x0)   execution time : 10.340 s
Press any key to continue.
```