

## 1 . Create a Vector class

Implement a Vector class that models a 2D vector in geometry.

The class should store two float values: x and y (representing vector components).

It must include:

- A constructor that takes x and y.
- A copy constructor that creates a deep copy of a Vector.
- An overloaded + operator to add two vectors.
- An overloaded = operator to assign one vector to another.

The screenshot displays the Visual Studio Code editor interface. The Explorer panel on the left shows a project named 'HOMEWORK 6 - VECTOR...' with a file 'C Vector.h' selected. The main editor window shows the implementation of the 'Vector' class in 'C Vector.h'. The code includes a constructor, a copy constructor, and overloaded operators for addition and assignment. The terminal window at the bottom shows the command prompt for 'garrettjackson@iPhone Homework 6 - Vector Class %' and 'garrettjackson@MacBookPro Homework 6 - Vector Class %', both displaying '\* History restored'.

```
1 class Vector
2 {
3     //Vector components
4     private:
5         float x;
6         float y;
7     public:
8         // Make a constructor with the initial values
9         Vector(float xCoordinate, float yCoordinate)
10        {
11            x = xCoordinate;
12            y = yCoordinate;
13        }
14
15        // Make a deep copy of the constructor
16        Vector(const Vector& copy)
17        {
18            // Copies the information of x and y into copy objects
19            x = copy.x;
20            y = copy.y;
21        }
22        // Overload the "+" operator to add 2 vectors
23        Vector operator+(const Vector& overload)
24        {
25            return Vector( x + overload.x, y + overload.y);
26        }
27
28        // Overload the "=" operator to assign one vector to another.
29        Vector& operator=(const Vector& overload)
30        {
31            //If statement to check if there is a self-assignment of an object
32            if(this == &overload)
33            {
34                return *this; // Does nothing and returns the object
35            }
36
37            // Copy the information from x and y
38            this->x = overload.x;
39            this->y = overload.y;
40
41            // Returns the copied data
42            return *this;
43        }
44    };
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

garrettjackson@iPhone Homework 6 - Vector Class %  
\* History restored

garrettjackson@MacBookPro Homework 6 - Vector Class %  
\* History restored

garrettjackson@Garretts-MacBook-Pro Homework 6 - Vector Class %

OUTLINE  
TIMELINE

main.cpp 0.1.1 0.0.0

Garrett Jackson (8 hours ago) Ln 38, Col 34 Spaces: 4 UTF-8 IE C++ Mac Prettier

class Vector

 $\{$

```
//Vector components
```

```
private:
```

```
    float x;
```

```
    float y;
```

```
public:
```

```
    // Make a constructor with the initial values
```

```
    Vector(float xCoordinate, float yCoordinate)
```

```
{
```

```
    x = xCoordinate;
```

```
    y = yCoordinate;
```

```
}
```

```
    // Make a deep copy of the contrstructor
```

```
    Vector(const Vector& copy)
```

```
{
```

```
    // Copies the information of x and y into copy objects
```

```
    x = copy.x;
```

```
    y = copy.y;
```

```
}
```

```
    // Overload the "+" operator to add 2 vectors
```

```
    Vector operator+(const Vector& overload)
```

```
{
```

```
    return Vector( x + overload.x, y + overload.y);
```

```
}
```

```
// Overload the "=" operator to assign one vector to another.
```

```
Vector& operator=(const Vector& overload)
```

```
{
```

```
    //If statement to check if there is a self-assignment of an object
```

```
    if(this == &overload)
```

```
    {
```

```
        return *this; // Does nothing and returns the object
```

```
    }
```

```
    // Copy the information from x and y
```

```
    this->x = overload.x;
```

```
    this->y = overload.y;
```

```
    // Returns the copied data
```

```
    return *this;
```

```
}
```

```
};
```