

```
1  #include <iostream>
2  #include <cmath>
3  #include "V2D.h"
4
5  using namespace std;
6
7
8  //////////////////////////////////////
9  //                                FRIEND FUNCTION                                //
10  //////////////////////////////////////
11
12  std::ostream & operator<<(std::ostream & os, const v2d & V) {
13      os << "[" << V.x << " , " << V.y << "];"
14      return os;
15  }
16
17
18  //////////////////////////////////////
19  //                                METHODS                                //
20  //////////////////////////////////////
21
22  // Constructor
23
24  v2d::v2d(double a, double b) {
25      x = a;
26      y = b;
27  }
28
29
30  // Another Constructor
31
32  v2d::v2d(const v2d & v) {
33      this->x = v.x;
34      this->y = v.y;
35  }
36
37
38  // Destructor
39
40  v2d::~v2d() { }
41
42
43  // Equality operator - Returns a reference to an object of v2d
44
45  v2d & v2d::operator=(const v2d & v) {
46      this->x = v.x;
47      this->y = v.y;
48      return *this;
49  }
50
51
52  // Addition operator - Returns a reference to an object of v2d
53
54  v2d & v2d::operator+(const v2d & v) {
55      this->x = this->x + v.x;
56      this->y = this->y + v.y;
57      return *this;
58  }
```

```
59
60
61 // Multiplication operator - Returns a double variable
62 // (vectorial multiplication)
63
64 double v2d::operator*(const v2d & v) {
65     return (this->x * v.x + this->y * v.y);
66 }
67
68
69 // Multiplication operator - Returns a reference to an object of v2d.
70 // Overloaded to provide scalar multiplication.
71
72 v2d& v2d::operator*(double k) {
73     x *= k;
74     y *= k;
75     return *this;
76 }
77
78
79 // Method that returns the module of the vector.
80
81 double v2d::length() {
82     return (sqrt(pow(x, 2) + pow(y, 2)));
83 }
```