

C++, Przeciążanie operatorów

Należy zaimplementować klasę `Vector2D`, reprezentującą wektor 2D o współrzędnych `double x` oraz `double y`.

Klasa `Vector 2D` zawiera:

- Konstruktor:
 - konstruktor domyślny (ustawia $x=y=0$),
 - konstruktor z dwoma argumentami.
- przeciążone operatory:

operator `+` : zwraca sumę dwóch wektorów
 $(x_1 + x_2, y_1 + y_2)$

operator `-` : zwraca różnicę dwóch wektorów
 $(x_1 - x_2, y_1 - y_2)$

operator `*` (`double`) : mnożenie wektora przez skalar
 $(x * k, y * k)$

operator `=` : zgodnie ze zdrowym rozsądkiem

operator `==` : zwraca `true`, jeśli współrzędne są takie same.

operator `<<` : wypisuje wektor w formacie: `(x, y)`

operator `[]` : zwraca `x` dla `[0]` oraz `y` dla `[1]`

Zaimplementuj następujące metody (nie jako operatory):

- **`double Length()`** : zwraca długość wektora
 $\sqrt{x^2 + y^2}$

Program powinien zawierać funkcję `main()` testującą działanie operatorów, np.:

```
Vector2D a(3.0, 4.0);
Vector2D b(1.0, -2.0);

Vector2D c = a + b;
cout << "c: " << c << "\n"; // (4, 2)
Vector2D d = a - b;
cout << "d: " << d << "\n"; // (2, 6)
Vector2D e = a * 2.0;
cout << "e: " << e << "\n"; // (6, 8)

bool equal = (a == Vector2(3, 4)); // true

cout << "equal: " << e << "\n";
cout << "Length of a: " << a.Length() << "\n";
```