

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи №4 з дисципліни
«Основи програмування – 2. Методології програмування»

«Перевантаження операторів»

Варіант 22

Виконав студент ІП-13, Музичук Віталій Андрійович
(шифр, прізвище, ім'я, по батькові)

Перевірила Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота 4

Перевантаження операторів

Мета – вивчити механізми створення класів з використанням перевантажених операторів (операцій)

Варіант 22

Завдання:

Розробити клас «Вектор у просторі», який задається координатами його кінця. Реалізувати для нього декілька конструкторів, геттери, метод обчислення довжини вектору. Перевантажити оператори «+» та «*» для знаходження суми і скалярного добутку векторів відповідно. Створити три вектори (M1, M2, M3), використовуючи різні конструктори. Визначити вектор M3 як суму векторів M1 та M2. Знайти довжину вектору M3, а також скалярний добуток векторів M1 та M3.

1. Виконання завдання на мові C++:

// Lab_4.cpp

```
#include "vector.h"

int main()
{
    cout << "Creating array M1" << endl;
    double* arr = new double[3];
    createVector(arr);
    Vector M1(arr[0], arr[1], arr[2]);
    printVector(M1);
    delete[] arr;

    double x;
    cout << "Creating array M2" << endl;
    cout << "Enter one coordinate: "; cin >> x;
    Vector M2(x);
    printVector(M2);

    Vector M3;
    cout << "Creating array M3" << endl << "M3 = M2 + M1 = ";
    M3 = M1 + M2;
    printVector(M3);
    cout << "Length of vector M3: " << M3.getLength() << endl;
    double scalarProduct = M1 * M3;

    cout << "Scalar product of M1 and M3 is: " << scalarProduct << endl;}
```

//vector.cpp

```
#include "vector.h"
```

```
Vector::Vector() : x(0), y(0), z(0) {}
```

```
Vector::Vector(double x1) : x(x1), y(x1), z(x1) {}
```

```
Vector::Vector(double x1, double y1, double z1) : x(x1), y(y1), z(z1) {}
```

```
Vector::Vector(Vector& vector1): x(vector1.x), y(vector1.y), z(vector1.z) {}
```

```
void printVector(Vector& vector1) {  
    printf_s("x: %.3f y: %.3f z: %.3f\n\n", vector1.getX(), vector1.getY(),  
vector1.getZ());  
}
```

```
double Vector::getLength() {  
    return sqrt(pow(x, 2) + pow(y, 2) + pow(z, 2));  
}
```

```
Vector Vector::operator+(Vector vector1) {  
    Vector res;  
    res.x = x + vector1.x;  
    res.y = y + vector1.y;  
    res.z = z + vector1.z;  
    return res;  
}
```

```
double operator*(Vector& vector1, Vector& vector2) {  
    return (vector1.x * vector2.x) + (vector1.y * vector2.y) + (vector1.z * vector2.z);  
}
```

```
void createVector(double* arr) {  
    string coordinates;  
    cout << "Enter vector's coordinates in format [x y z]: ";  
    getline(cin, coordinates);  
  
    int pos;  
    for (int i = 0; i < 3; i++)  
    {  
        if (i == 2)  
            arr[i] = stod(coordinates);  
        else {  
            pos = coordinates.find(' ');  
            arr[i] = stod(coordinates.substr(0, pos));  
            coordinates.erase(0, pos + 1);  
        }  
    }  
}
```

// vector.h


```
#pragma once

#include <iostream>
#include <math.h>
#include <string>
using namespace std;

class Vector {
private:
    double x;
    double y;
    double z;
public:
    Vector();
    Vector(double);
    Vector(double, double, double);
    Vector(Vector& vector1);
    double getX() { return x; }
    double getY() { return y; }
    double getZ() { return z; }
    double getLength();
    Vector operator+(Vector);
    friend double operator*(Vector& vector1, Vector& vector2);
};

void createVector(double* arr);
void printVector(Vector& vector1);
```

Тестування програми:


 Microsoft Visual Studio Debug Console

```
Creating array M1
Enter vector's coordinates in format [x y z]: 10 15 3
x: 10.000 y: 15.000 z: 3.000

Creating array M2
Enter one coordinate: 9
x: 9.000 y: 9.000 z: 9.000

Creating array M3
M3 = M2 + M1 = x: 19.000 y: 24.000 z: 12.000

Length of vector M3: 32.8786
Scalar product of M1 and M3 is: 586
```

 Microsoft Visual Studio Debug Console

```
Creating array M1
Enter vector's coordinates in format [x y z]: 3.5 10 7
x: 3.500 y: 10.000 z: 7.000

Creating array M2
Enter one coordinate: 2.3
x: 2.300 y: 2.300 z: 2.300

Creating array M3
M3 = M2 + M1 = x: 5.800 y: 12.300 z: 9.300

Length of vector M3: 16.4748
Scalar product of M1 and M3 is: 208.4
```

D:\KPI\Programming\Second semester\C++\Debug\Lab_4.exe (pr

Висновок: Під час виконання лабораторної роботи я вивчив особливості створення класів з використанням перевантажених операторів (операцій) на прикладі мови C++. Результатом виконання лабораторної роботи є програма, основним завданням якої є створення класу вектору та ініціалізація трьох його об'єктів. Після тестування програм можна зробити висновок, що вони справляються із поставленою задачею.