

Міністерство освіти і науки України

КПІ ім. Ігоря Сікорського

Кафедра ІІІ

ЗВІТ

з виконання лабораторної роботи № 3

з кредитного модуля

“Основи програмування-2. Методології програмування”

Варіант № 1

Виконав:

студент 1-го курсу

гр. ІІ-22 ФІОТ

Андреева Уляна Андріївна

Київ 2023

ПОСТАНОВКА ЗАДАЧІ

1. Розробити клас "Вектор", який задається полярними координатами (довжина і кут). Реалізувати для нього декілька конструкторів, геттери, метод повороту вектору на будь-який кут. Перевантажити оператори "+" та "/" для знаходження суми векторів і "зменшення" вектору відповідно. Створити три вектори (B1, B2, B3), використовуючи різні конструктори. Повернути вектор B3 на 45° , вектор B2 "зменшити" у 2 рази. Визначити вектор B1 як суму змінених векторів B2 та B3.

Код програми

main.cpp

```
#include <iostream>
#include <cmath>
#include "Functions.h"
using namespace std;

int main() {
    Vector v1, v2, v3;

    cout<<"\t\t\t\t\tINITIAL DATA:"<< endl;
    cout << "Vector 1 is initialized with empty constructor." << std::endl;
    cout << "Enter vector 2:" << endl;
    v2 = CreateVector();
    cout << "Enter vector 3:" << endl;
    v3 = CreateVector();
    cout<<"-----"<<endl;
    cout<<"\t\t\t\t\tENTERED VECTORS: \n" <<v1<<"\n"<<v2<<"\n"<<v3<<endl;
    cout<<"-----"<<endl;
    cout<<"\t\t\t\t\tINTERIM RESULTS:"<< endl;
    v3.rotate(M_PI / 4);
    cout << "Vector 3 after rotation: " << v3<< endl;
    v2 = v2 / 2;
    cout<<"Vector 2 after dividing : "<< v2<< endl;
    v1 = v2 + v3;
    cout<<"Vector 1 after summing the first and the second one : "<< v1<<
endl;
    cout<<"-----"<<endl;
    cout<<"\t\t\t\t\tFINAL RESULTS:" << endl;
    cout << "Vector 1: " << v1 << endl;
    cout << "Vector 2: " << v2 << endl;
    cout << "Vector 3: " << v3 << endl;
    return 0;
}
```

Functions.cpp

```
#include <iostream>
#include "Functions.h"

Vector CreateVector() {
    double len, ang;
    char ch,error;
```

```

        symbolCheckingWithMinus(error, ch, len, "Enter the length: ");
        symbolChecking(error, ch, ang, "Enter the angle (in radians): ");
        Vector vector(len, ang);

        return vector;
    }

void symbolChecking(char error, char &ch, double &x1, std::string text) {
    do
    {
        error=0;
        std::cout << text;
        scanf("%lf%c", &x1, &ch);
        if (ch!='\n')
        {
            error = 1;
            fflush(stdin);
            printf("Error inputting\n");
        }
        ch = 0;

    }
    while(error);
}

void symbolCheckingWithMinus(char error, char &ch, double &x1, std::string
text) {
    do
    {
        error=0;
        std::cout << text;
        scanf("%lf%c", &x1, &ch);
        if (ch!='\n' || (x1<0))
        {
            error = 1;
            fflush(stdin);
            printf("Error inputting\n");
        }
        ch = 0;

    }
    while(error);
}

```

Functions.h

```

#ifndef LABWORK4_FUNCTIONS_H
#define LABWORK4_FUNCTIONS_H

#endif //LABWORK4_FUNCTIONS_H

#include "Vector.h"

Vector CreateVector();
void symbolChecking(char error, char &ch, double &x1, std::string text);
void symbolCheckingWithMinus(char error, char &ch, double &x1, std::string
text);

```

Vector.cpp

```
#include <iostream>
#include "cmath"
#include "Vector.h"

Vector::Vector() {
    length = 0;
    angle = 0;
}

Vector::Vector(double len, double ang) {
    length = len;
    angle = ang;
}

double Vector::getLength() const {
    return length;
}

double Vector::getAngle() const {
    return angle;
}

void Vector::setLength(double len) {
    length = len;
}

void Vector::setAngle(double ang) {
    angle = ang;
}

void Vector::rotate(double ang) {
    angle += ang;
}

Vector Vector::operator+(const Vector &v) const {
    double x = length * cos(angle) + v.length * cos(v.angle);
    double y = length * sin(angle) + v.length * sin(v.angle);
    double len = sqrt(x*x + y*y);
    double ang = atan2(y, x);
    return Vector(len, ang);
}

Vector Vector::operator/(double n) const {
    return Vector(length / n, angle);
}

std::ostream& operator<<(std::ostream& os, const Vector& v) {
    os << "Vector: (" << v.getLength() << ", " << v.getAngle() << ")";
    return os;
}
```

Vector.h

```
#ifndef LABWORK4_VECTOR_H
#define LABWORK4_VECTOR_H

class Vector {
private:
    double length;
    double angle;

public:
    Vector();
    Vector(double len, double ang);

    double getLength() const ;

    double getAngle() const ;

    void setLength(double len);

    void setAngle(double ang) ;

    void rotate(double ang);

    Vector operator+(const Vector &v) const ;

    Vector operator/(double n) const ;

    friend std::ostream& operator<<(std::ostream& os, const Vector& v);
};

#endif //LABWORK4_VECTOR_H
```

Результати програми

```
/Users/mac/CLionProjects/labwork4/cmake-build-debug/labwork4
    INITIAL DATA:
Vector 1 is initialized with empty constructor.
Enter vector 2:
Enter the length: 2
Enter the angle (in radians): 0.45
Enter vector 3:
Enter the length: 3.5
Enter the angle (in radians): 2
-----
    ENTERED VECTORS:
Vector: (0, 0)
Vector: (2, 0.45)
Vector: (3.5, 2)
-----
    INTERIM RESULTS:
Vector 3 after rotation: Vector: (3.5, 2.7854)
Vector 2 after dividing : Vector: (1, 0.45)
Vector 1 after summing the first and the second one : Vector: (2.89901, 2.53382)
-----
    FINAL RESULTS:
Vector 1: Vector: (2.89901, 2.53382)
Vector 2: Vector: (1, 0.45)
Vector 3: Vector: (3.5, 2.7854)

Process finished with exit code 0
```

Лінк на репозиторій у GitHub:

<https://github.com/Uliana200407/CppProjects-.git>