

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 10

Виконав студент ІП-15 Закірова Олександра Володимирівна
Перевірів Вечерковська Анастасія Сергіївна

Київ 2022

Лабораторна робота 4

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

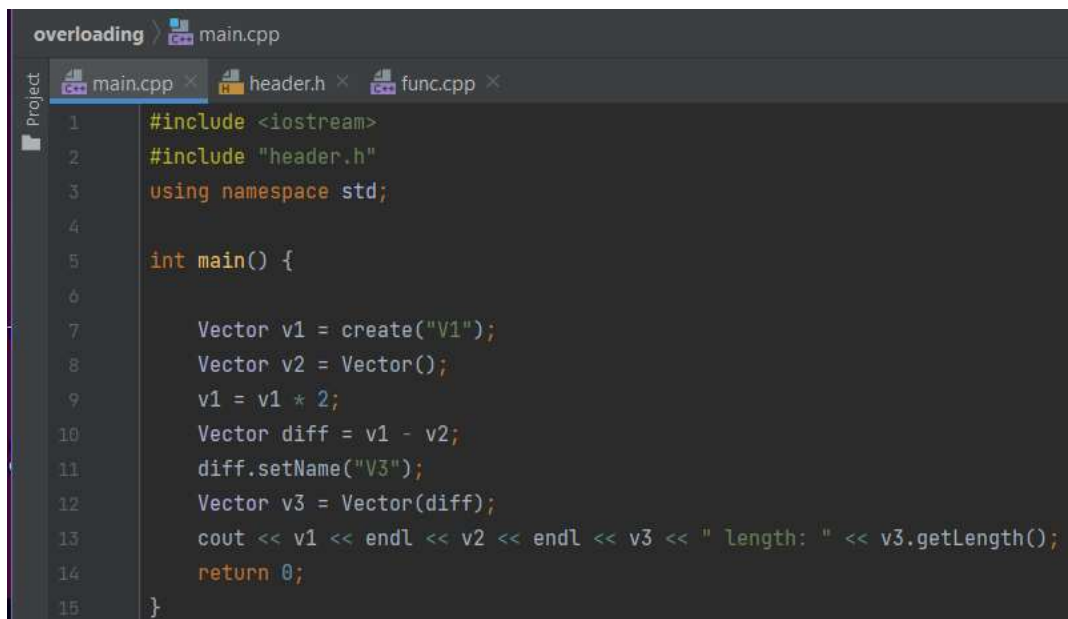
Індивідуальне завдання

Варіант 10

10. Розробити клас "Вектор на площині", який задається координатами його кінця. Реалізувати для нього декілька конструкторів, геттери, метод обчислення довжини вектору. Перевантажити оператори "-" та "*" для знаходження різниці векторів і "збільшення" вектору відповідно. Створити три вектори (V1, V2, V3), використовуючи різні конструктори. Вектор V1 "збільшити" у 2 рази. Визначити вектор V3 як різницю зміненого вектору V1 та вектору V2. Знайти довжину вектору V3.

Виконання C++

Main.cpp



```
overloading > main.cpp
Project main.cpp x header.h x func.cpp x
1  #include <iostream>
2  #include "header.h"
3  using namespace std;
4
5  int main() {
6
7      Vector v1 = create("V1");
8      Vector v2 = Vector();
9      v1 = v1 * 2;
10     Vector diff = v1 - v2;
11     diff.setName("V3");
12     Vector v3 = Vector(diff);
13     cout << v1 << endl << v2 << endl << v3 << " length: " << v3.getLength();
14     return 0;
15 }
```

Header.h

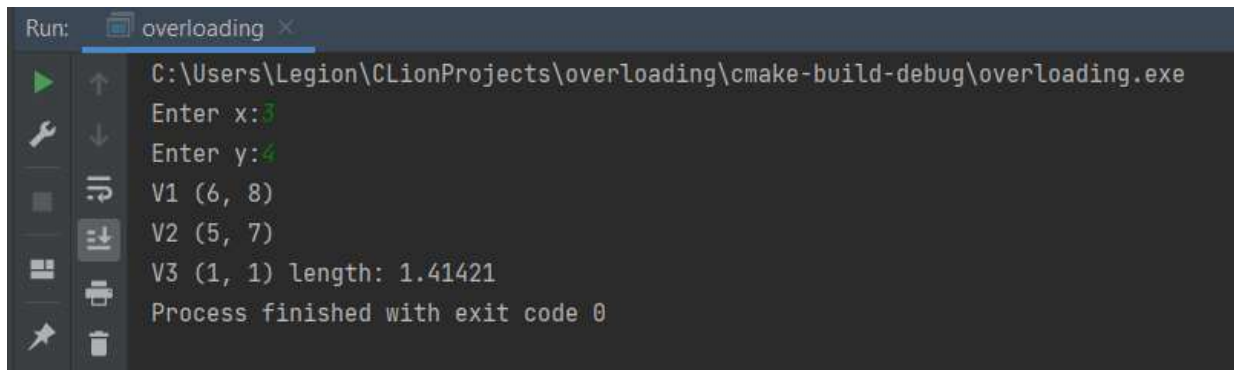
```
overloading > header.h
main.cpp x header.h x func.cpp x
Project
3  #include <cmath>
4  #include <utility>
5
6  using namespace std;
7
8  class Vector {
9
10     private:
11         string name = "V3";
12         int x;
13         int y;
14         double length = getLength();
15
16     public:
17         Vector(string name, int x, int y);
18         Vector();
19         Vector(int x, int y);
20         friend ostream &operator<<(ostream &os, const Vector &vector);
21         Vector operator-(const Vector &vector) const;
22         Vector operator*(int) const;
23         double getLength() const;
24         int getX() const;
25         int getY() const;
26         string getName();
27         void setName(const string &string);
28     };
29
30     Vector create(const string &name);
Structure
```

Func.cpp

```
overloading > func.cpp
main.cpp x header.h x func.cpp x
1 #include "header.h"
2
3 Vector::Vector(string name, int x, int y) : name(std::move(name)), x(x), y(y), length(getLength()) {}
4
5 Vector::Vector() {
6     name = "V2";
7     x = 5;
8     y = 7;
9     length = getLength();
10 }
11
12 Vector::Vector(int x, int y) : x(x), y(y) {}
13
14 ostream &operator<<(ostream &os, const Vector &vector) {
15     os << vector.name << " (" << vector.x << ", " << vector.y << ")";
16     return os;
17 }
18
19 Vector Vector::operator-(const Vector &vector) const {
20     int vx = x - vector.x;
21     int vy = y - vector.y;
22     return {vx, vy};
23 }
24
25 Vector Vector::operator*(int multiplier) const {
26     int vx = x * multiplier;
27     int vy = y * multiplier;
28     return {name, vx, vy};
29 }
```

```
overloading > func.cpp
main.cpp x header.h x func.cpp x
30
31 double Vector::getLength() const {
32     return sqrt((x * x) + (y * y));
33 }
34
35 int Vector::getX() const {
36     return x;
37 }
38
39 int Vector::getY() const {
40     return y;
41 }
42 |
43 string Vector::getName() {
44     return name;
45 }
46
47 void Vector::setName(const string &string) {
48     name = string;
49 }
50
51 Vector create(const string &name) {
52     int x, y;
53     cout << "Enter x:";
54     cin >> x;
55     cout << "Enter y:";
56     cin >> y;
57     return {name, x, y};
58 }
59
```

Результат виконання програми:



```
Run: overloading x
C:\Users\Legion\CLionProjects\overloading\cmake-build-debug\overloading.exe
Enter x:3
Enter y:4
V1 (6, 8)
V2 (5, 7)
V3 (1, 1) length: 1.41421
Process finished with exit code 0
```

Висновки

Протягом лабораторної роботи було розглянуто роботу з класами та їх об'єктами, різними видами конструкторів та перевантаженням операцій, використано отримані навички під час написання програм. В результаті роботи було створену програму, яка створює три вектори різними видами конструкторів, за допомогою перевантажених операцій із перших двох шукає третій.