

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 № 3 з дисципліни
«Основи програмування - 2.
Модульне програмування»

«Класи та об'єкти»

Варіант 5

Виконав студент

ІП-15, Буяло Дмитро Олександрович
(шифр, прізвище, ім'я, по батькові)

Перевірів

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

Київ 2022

Лабораторна робота 3

Класи та об'єкти

Мета – вивчити механізми створення та використання класів й об'єктів.

Індивідуальне завдання

Варіант 5

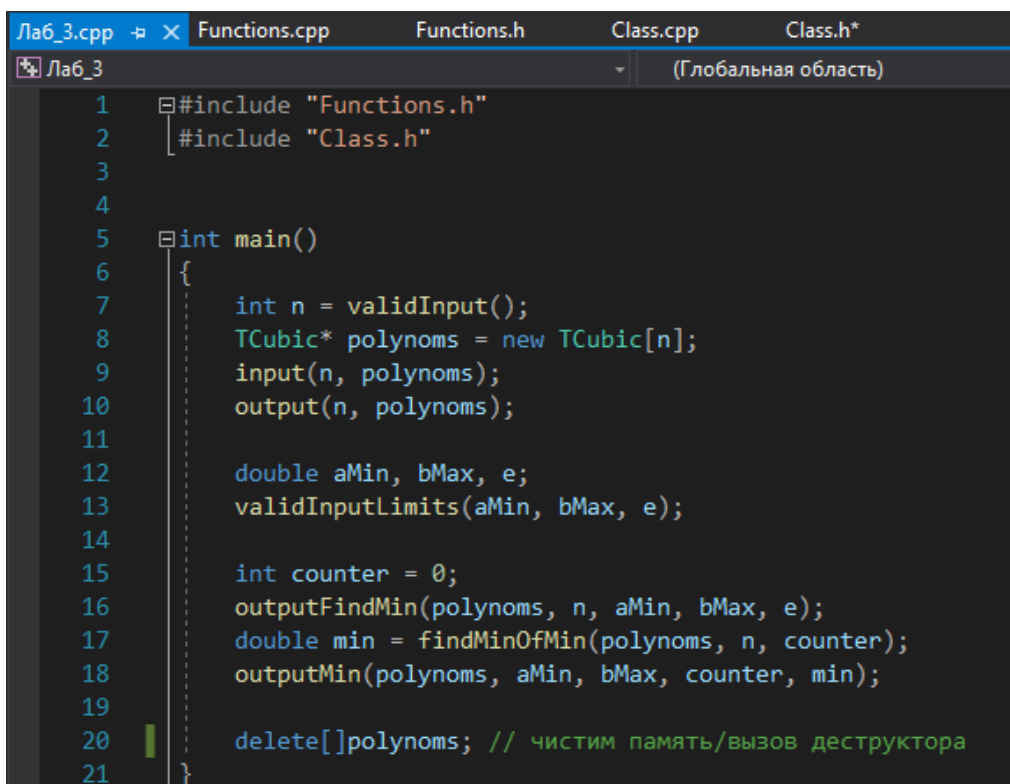
Завдання

Розробити клас «кубічний многочлен»

(многочлен виду $a_3x^3 + a_2x^2 + a_1x + a_0$),

який заданий своїми коефіцієнтами. Створити масив об'єктів даного класу. Визначити многочлен, який приймає найменше мінімальне значення на введеному користувачем відрізьку $[a; b]$ із заданою точністю ε (використати метод простого перебору).

1. Код на C++



```
Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3
1  #include "Functions.h"
2  #include "Class.h"
3
4
5  int main()
6  {
7      int n = validInput();
8      TCubic* polynoms = new TCubic[n];
9      input(n, polynoms);
10     output(n, polynoms);
11
12     double aMin, bMax, e;
13     validInputLimits(aMin, bMax, e);
14
15     int counter = 0;
16     outputFindMin(polynoms, n, aMin, bMax, e);
17     double min = findMinOfMin(polynoms, n, counter);
18     outputMin(polynoms, aMin, bMax, counter, min);
19
20     delete[] polynoms; // чистим пам'ять/вызов деструктора
21 }
```

```

Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3      (Глобальна область)

1  #include "Functions.h"
2  #include "Class.h"
3
4
5  void invalidInput(double& input) { // абсолютна валідація некоректного вводу
6      while (!cin) {
7          cin.clear();
8          cin.ignore(64, '\n');
9          cout << "Your input is wrong, try again: "; cin >> input;
10     }
11 }
12
13 void invalidInput(int& input) { // перегружаємо для інта
14     while (!cin || input < 1) {
15         cin.clear();
16         cin.ignore(64, '\n');
17         cout << "Your input is wrong, try again: "; cin >> input;
18     }
19 }
20
21 int validInput() { // валідація вводу
22     int n;
23     cout << "Enter the number of polynoms: "; cin >> n;
24     invalidInput(n);
25     cout << endl;
26     return n;
27 }
28
29 void input(int n, TCubic* polynoms) { // вводим коефіцієнти і заносимо в масив об'єктів
30     double coeff[4];
31     for (int i = 0; i < n; i++) {
32         for (int i = 0; i < 4; i++) {
33             cout << "Enter coefficient a" << 3 - i << ": "; cin >> coeff[i];
34             invalidInput(coeff[i]);
35         }
36         polynoms[i] = TCubic(coeff[0], coeff[1], coeff[2], coeff[3]);
37         cout << endl;
38     }
39 }
40
41 void output(int n, TCubic* polynoms) { // выводим содержание массива объектов
42     for (int i = 0; i < n; i++) {
43         cout << "Your " << i + 1 << " polynom: ";
44         polynoms[i].printEquation(); cout << endl;
45     }
46     cout << endl;
47 }

```

```

Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3
(Глобальная область)
49 void validInputLimits(double& min, double& max, double& e) { // ввод границ и точности
50     cout << "Enter the minimum value of the segment a: "; cin >> min;
51     invalidInput(min);
52     cout << "Enter the maximum value of the segment b: "; cin >> max;
53     invalidInput(max);
54     do {
55         cout << "Enter accuracy e: "; cin >> e;
56         invalidInput(e);
57     } while (e <= 0);
58     double tmp;
59     if (min > max) {
60         tmp = min;
61         min = max;
62         max = tmp;
63     }
64     cout << endl;
65 }
66
67 // промежуточный вывод минимумов всех уравнений на отрезке
68 void outputFindMin(TCubic* polynoms, int n, double min, double max, double e) {
69     for (int i = 0; i < n; i++) {
70         polynoms[i].findMin(min, max, e);
71         cout << "The minimum value of " << i + 1 << " polynomial ";
72         polynoms[i].printEquation();
73         cout << " is " << polynoms[i].getMin(); cout << endl;
74     }
75 }
76
77 double findMinOfMin(TCubic* polynoms, int n, int& counter) { // находим минимальное из всех минимальных
78     double min;
79     min = polynoms[0].getMin();
80     for (int i = 1; i < n; i++) {
81         if (polynoms[i].getMin() < min) {
82             min = polynoms[i].getMin();
83             counter = i;
84         }
85     }
86     return min;
87 }
88
89 // выводим окончательный ответ
90 void outputMin(TCubic* polynoms, double aMin, double bMax, int count, double min) {
91     cout << "\nThe minimum value on [" << aMin << "; " << bMax << "] has " << count + 1 << " polynomial" << endl;
92     polynoms[count].printEquation(); cout << " = " << min; cout << endl;
93 }

```

```

Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3
1  #pragma once
2  #include "Class.h"
3
4
5  void invalidInput(double&);
6  void invalidInput(int&);
7  int validInput();
8  void input(int, TCubic*);
9  void output(int, TCubic*);
10 void validInputLimits(double&, double&, double&);
11 void outputFindMin(TCubic*, int, double, double, double);
12 double findMinOfMin(TCubic*, int, int&);
13 void outputMin(TCubic*, double, double, int, double);

```

```

Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3
1  #pragma once
2  #include <iostream>
3  #include <iomanip>
4
5  using namespace std;
6
7  class TCubic {
8  public:
9      double a, b, c, d, min;
10     /* ... */
11     double getMin();
12     void setMin(double m);
13     void findMin(double, double, double);
14     void printEquation();
15     // конструктор по умолчанию (без параметров)
16     TCubic() { a = b = c = d = 0; } // выделение памяти под динамический массив
17     TCubic(double a3, double a2, double a1, double a0) : a(a3), b(a2), c(a1), d(a0) {} // конструктор инициализации
18     ~TCubic() {}
19 };

```

```

Лаб_3.cpp  Functions.cpp  Functions.h  Class.cpp  Class.h*
Лаб_3  TCubic  printEquation()

1  #include "Class.h"
2
3  /* ... */
27 double TCubic::getMin() {
28     return min;
29 }
30 void TCubic::setMin(double m) {
31     min = m;
32 }
33 void TCubic::findMin(double aMin, double bMax, double e) { // находим минимум функции
34     double min, current, newE;
35     newE = e / 2;
36     min = a * pow(aMin, 3) + b * pow(aMin, 2) + c * aMin + d;
37     for (double x = aMin + e; x <= bMax; x += newE) {
38         current = a * pow(x, 3) + b * pow(x, 2) + c * x + d;
39         //13.48651358952 -> 13.4865 при e=0.0001 - понятие точности в этой задаче
40         if (current < min) min = current;
41         min = round(min * (1 / e)) / (1 / e);
42     }
43     setMin(min);
44 }
45 void TCubic::printEquation() { // красивый вывод целого уравнения
46     if (b < 0) {
47         if (c < 0) {
48             if (d < 0) cout << a << "x^3 - " << abs(b) << "x^2 - " << abs(c) << "x - " << abs(d);
49             else cout << a << "x^3 - " << abs(b) << "x^2 - " << abs(c) << "x + " << d;
50         }
51         else {
52             if (d < 0) cout << a << "x^3 - " << abs(b) << "x^2 + " << c << "x - " << abs(d);
53             else cout << a << "x^3 - " << abs(b) << "x^2 + " << c << "x + " << d;
54         }
55     }
56     else {
57         if (c < 0) {
58             if (d < 0) cout << a << "x^3 + " << b << "x^2 - " << abs(c) << "x - " << abs(d);
59             else cout << a << "x^3 + " << b << "x^2 - " << abs(c) << "x + " << d;
60         }
61         else {
62             if (d < 0) cout << a << "x^3 + " << b << "x^2 + " << c << "x - " << abs(d);
63             else cout << a << "x^3 + " << b << "x^2 + " << c << "x + " << d;
64         }
65     }
66 }

```

2. Результат виконання на C++

```
Enter the number of polynoms: 9
Enter coefficient a3: 1
Enter coefficient a2: 1
Enter coefficient a1: -4
Enter coefficient a0: -4

Enter coefficient a3: 1
Enter coefficient a2: -5
Enter coefficient a1: 8
Enter coefficient a0: -4

Enter coefficient a3: -1
Enter coefficient a2: -4.8
Enter coefficient a1: -5
Enter coefficient a0: 1.2

Enter coefficient a3: 4
Enter coefficient a2: -3
Enter coefficient a1: 0
Enter coefficient a0: -1

Enter coefficient a3: 10
Enter coefficient a2: -9
Enter coefficient a1: -3
Enter coefficient a0: 2

Enter coefficient a3: 4
Enter coefficient a2: -19
Enter coefficient a1: 19
Enter coefficient a0: 6

Enter coefficient a3: 0
Enter coefficient a2: 4
Enter coefficient a1: -2
Enter coefficient a0: -1

Enter coefficient a3: 0
Enter coefficient a2: 0
Enter coefficient a1: 1
Enter coefficient a0: -2

Enter coefficient a3: 0
Enter coefficient a2: 0
Enter coefficient a1: 0
Enter coefficient a0: 1.5

Your 1 polynom: 1x^3 + 1x^2 - 4x - 4
Your 2 polynom: 1x^3 - 5x^2 + 8x - 4
Your 3 polynom: -1x^3 - 4.8x^2 - 5x + 1.2
Your 4 polynom: 4x^3 - 3x^2 + 0x - 1
Your 5 polynom: 10x^3 - 9x^2 - 3x + 2
Your 6 polynom: 4x^3 - 19x^2 + 19x + 6
Your 7 polynom: 0x^3 + 4x^2 - 2x - 1
Your 8 polynom: 0x^3 + 0x^2 + 1x - 2
Your 9 polynom: 0x^3 + 0x^2 + 0x + 1.5

Enter the minimum value of the segment a: -2
Enter the maximum value of the segment b: 3
Enter accuracy e: 0.001

The minimum value of 1 polynom 1x^3 + 1x^2 - 4x - 4 is -6.065
The minimum value of 2 polynom 1x^3 - 5x^2 + 8x - 4 is -48
The minimum value of 3 polynom -1x^3 - 4.8x^2 - 5x + 1.2 is -83.97
The minimum value of 4 polynom 4x^3 - 3x^2 + 0x - 1 is -45
The minimum value of 5 polynom 10x^3 - 9x^2 - 3x + 2 is -108
The minimum value of 6 polynom 4x^3 - 19x^2 + 19x + 6 is -140
The minimum value of 7 polynom 0x^3 + 4x^2 - 2x - 1 is -1.25
The minimum value of 8 polynom 0x^3 + 0x^2 + 1x - 2 is -4
The minimum value of 9 polynom 0x^3 + 0x^2 + 0x + 1.5 is 1.5

The minimum value on [-2; 3] has 6 polynomial
4x^3 - 19x^2 + 19x + 6 = -140
```

3. Висновок

Під час виконання третьої лабораторної роботи, розглянули на практиці створення та роботу з класами та об'єктами. В результаті було створена програма, яка знаходить найменше мінімальне значення на введеному користувачем відрізьку з заданою точністю.