



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа №9
по курсу «Языки и методы программирования»
«Перегрузка операций»

Студент группы ИУ9-22Б Лавров Р. Д.

Преподаватель Посевин Д. П.

Москва 2025

1 Задание

Вариант 12: $\text{Curve}\langle T \rangle$ – кривая на плоскости, заданная функцией $y = f(x)$, где x и y – числа с плавающей точкой типа T . Конструктор кривой принимает булевское значение i , в зависимости от этого значения, порождает либо кривую $y = \sin x$, либо кривую $y = \cos x$. Операции:

1. «+» – сумма двух кривых: $y = y_1 + y_2 = f_1(x) + f_2(x)$;
2. «-» - разность двух кривых: $y = y_1 - y_2 = f_1(x) - f_2(x)$;
3. «*» - умножение кривой на число: $y = ky_1 = kf_1(x)$;
4. «-» - изменение знака (унарный минус): $y = -y_1 = -f_1(x)$;
5. «!» – дифференцирование: $y = dy_1 / dx$;
6. «()» – вычисление y для указанного x .

Вариант 49: $\text{ApproxNum}\langle T \rangle$ - число вида $a + kd$, где a и k имеют тип T , а d – неизвестное бесконечно малое число. Сложение, умножение на число типа T и сравнение чисел $\text{ApproxNum}\langle T \rangle$ определяется правилами:

1. $(a_1 + k_1d) + (a_2 + k_2d) == a_1 + a_2 + (k_1 + k_2)d$;
2. $c(a + kd) == ca + ckd$;
3. $a_1 + k_1d < a_2 + k_2d$ тогда и только тогда, когда либо $a_1 < a_2$, либо $(a_1 = a_2)$ or $(k_1 < k_2)$

Операции, перегружаемые для $\text{ApproxNum}\langle T \rangle$:

1. «*=» - домножение на число типа T ;
2. «+=», «-=» - прибавление и вычитание числа типа $\text{ApproxNum}\langle T \rangle$;
3. «+», «-» - сложение и вычитание двух чисел типа $\text{ApproxNum}\langle T \rangle$;
4. «*» – умножение числа типа $\text{ApproxNum}\langle T \rangle$ на число типа T , а также умножение числа типа T на число типа $\text{ApproxNum}\langle T \rangle$;
5. «==», «!=», «<», «<=», «>», «>=» – сравнение чисел типа $\text{ApproxNum}\langle T \rangle$.

2 Результаты

Исходный код 1–4.

Листинг 1 — Curve.cpp (12 вариант)

```
1 #include <cmath>
2 #include <iostream>
3
4 using namespace std;
5
6 template<typename T>
7 class Curve {
8 private:
9     T sinRatio;
10    T cosRatio;
11 public:
12    Curve(bool flag) {
13        if (flag) {
14            sinRatio = 1;
15            cosRatio = 0;
16        } else {
17            cosRatio = 1;
18            sinRatio = 0;
19        }
20    }
21
22    Curve(T sinK, T cosK) : sinRatio(sinK), cosRatio(cosK) {}
23
24    Curve operator+(Curve& other){
25        auto new_sinRatio = this->sinRatio + other.sinRatio;
26        auto new_cosRatio = this->cosRatio + other.cosRatio;
27        return Curve(new_sinRatio, new_cosRatio);
28    }
29
30    Curve operator-(Curve& other){
31        auto new_sinRatio = this->sinRatio - other.sinRatio;
32        auto new_cosRatio = this->cosRatio - other.cosRatio;
33        return Curve(new_sinRatio, new_cosRatio);
34    }
35
36    Curve operator*(T k){
37        auto new_sinRatio = k*this->sinRatio;
38        auto new_cosRatio = k*this->cosRatio;
39        return Curve(new_sinRatio, new_cosRatio);
40    }
41
42    Curve operator-(){
43        auto new_sinRatio = -this->sinRatio;
44        auto new_cosRatio = -this->cosRatio;
45        return Curve(new_sinRatio, new_cosRatio);
46    }
47
48    T operator()(T x){
49        return sinRatio * sin(x) + cosRatio * cos(x);
50    }
```

Листинг 2 — Curve.cpp (12 вариант) (продолжение)

```

1  Curve operator!(){
2      auto new_sinRatio = -this->cosRatio;
3      auto new_cosRatio = this->sinRatio;
4      return Curve(new_sinRatio, new_cosRatio);
5  }
6
7  void show() {
8      cout << sinRatio << "sin(x) + " << cosRatio << "cos(x)" << endl;
9  }
10 };
11
12 int main(){
13     Curve<float> curve1(true);
14     Curve<float> curve2(false);
15
16     auto curve3 = (curve1 * 2) + curve2;
17
18     curve3.show();
19     curve3 = !curve3;
20     curve3.show();
21
22     cout << curve3(1.5);
23 }

```

Листинг 3 — ApproxNum.cpp (49 вариант)

```

1  #include <cmath>
2  #include <iostream>
3
4  using namespace std;
5
6  template<typename T>
7  class ApproxNum {
8  private:
9      T a;
10     T k;
11 public:
12     ApproxNum(T a, T k) : a(a), k(k) {}
13
14     ApproxNum operator*=(T k) {
15         this->a = k * this->a;
16         this->k = k * this->k;
17         return *this;
18     }
19
20     ApproxNum operator+=(ApproxNum& other) {
21         this->a = this->a + other.a;
22         this->k = this->k + other.k;
23         return *this;
24     }

```

Листинг 4 — ApproxNum.cpp (49 вариант) (продолжение)

```

1  ApproxNum operator=(ApproxNum& other) {
2      this->a = this->a - other.a;
3      this->k = this->k - other.k;
4      return *this;
5  }
6
7  ApproxNum operator+(ApproxNum& other) {
8      auto new_a = this->a + other.a;
9      auto new_k = this->k + other.k;
10     return ApproxNum(new_a, new_k);
11 }
12
13 ApproxNum operator-(ApproxNum& other) {
14     auto new_a = this->a - other.a;
15     auto new_k = this->k - other.k;
16     return ApproxNum(new_a, new_k);
17 }
18
19 ApproxNum operator*(T k) {
20     auto new_a = k * this->a;
21     auto new_k = k * this->k;
22     return ApproxNum(new_a, new_k);
23 }
24
25 bool operator==(ApproxNum& other) {
26     if (this->a == other.a && this->k == other.k){
27         return true;
28     }
29     return false;
30 }
31
32 bool operator!=(ApproxNum& other) {
33     return !(this == other);
34 }
35
36 bool operator<(ApproxNum& other) {
37     if (this->a < other.a || (this->a == other.a && this->k < other.k)){
38         return true;
39     }
40     return false;
41 }
42
43 bool operator<=(ApproxNum& other) {
44     return this < other || this == other;
45 }
46
47 bool operator>(ApproxNum& other) {
48     return !(this < other) && !(this == other);
49 }
50
51 bool operator>=(ApproxNum& other) {return this > other || this == other; }
52 void show(){ cout << this->a << "+" << this->k << "d" << endl; }
53 };
54
55 int main(){
56     ApproxNum<int> n1(10, 5);
57     ApproxNum<int> n2(6, 2);
58     n1 *= 2;
59     n1.show();
60     auto n3 = n1 + n2;
61     n3.show();
62 }

```

```
● veter.ok77@MacBook-Pro-Rodion lab9 % ./curve  
2sin(x) + 1cos(x)  
-1sin(x) + 2cos(x)  
-0.856021%  
○ veter.ok77@MacBook-Pro-Rodion lab9 %
```

Рис. 1 — Результат работы

```
● veter.ok77@MacBook-Pro-Rodion lab9 % ./ApproxNum  
20+106  
26+126  
○ veter.ok77@MacBook-Pro-Rodion lab9 %
```

Рис. 2 — Результат работы

3 Вывод

Я научился делать перегрузку операций