МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСТИТЕТ)

ЛАБОРАТОРНАЯ РАБОТА №2

по курсу “Объектно-ориентированное программирование”
I семестр, 2021/22 учебный год

Студент: *Степанов Данила Михайлович, группа М8О-207Б-20*
Преподаватель: *Дорохов Евгений Павлович, каф. 806*

Задание:

Разработать программу на языке C++ согласно варианту задания. Программа должна получать данные из стандартного ввода и выводить данные в стандартный вывод. Реализовать пользовательский литерал для работы с константами объектов созданного класса.

Вариант №4:

Реализовать класс FazzyNumber для работы с нечеткими числами, которые представляются тройками чисел ($x – e_l, x, x + e_r$). Для чисел $A = (A – a_l, A, A + a_r)$ и $B = (B – b_l, B, B + b_r)$ арифметические операции выполняются по следующим формулам:

- сложение $A + B = (A + B – a_l – b_l, A + B, A + B + a_r + b_r)$;
- вычитание $A – B = (A – B – a_l – b_l, A – B, A – B + a_r + b_r)$;
- умножение $A \cdot B = (A \cdot B – B \cdot a_l – A \cdot b_l + a_l \cdot b_l, A \cdot B, A \cdot B + B \cdot a_l + A \cdot b_l + a_l \cdot b_l)$;
- обратное число $A = (1 / (A + a_r), 1 / A, 1 / (A – a_l)), A > 0$;
- деление $A / B = ((A – a_l) / (B + b_r), A / B, (A + a_r) / (B – b_l)), B > 0$;

Считать $e_l = e_r$, то есть число представлено парой $<x, e>$.

Реализовать операции сравнения по x.

Описание программы:

Исходный код разделён на 3 файла:

- FazzyNumber.h – описание класса FazzyNumber
- FazzyNumber.cpp – реализация класса FazzyNumber
- main.cpp – основная программа

Дневник отладки:

Проблем не возникло.

Тестирование программы:

```
Sum of fazzy numbers:(15, 18, 21)
Difference of fazzy numbers: (-7, -4, -1)
Composition of fazzy numbers: (54, 77, 104)
Quotient of fazzy numbers: (0.461538, 0.636364, 0.888889)
Opposite fazzy number: (0.125, 0.142857, 0.166667)
First number < than second one

Sum of fazzy numbers:(8.5, 9.45, 10.4)
```

```
Difference of fazzy numbers: (-8.4, -7.45, -6.5)
Composition of fazzy numbers: (4, 8.45, 13.35)
Quotient of fazzy numbers: (0.0561798, 0.118343, 0.1875)
Opposite fazzy number: (0.666667, 1, 2)
First number < than second one

Sum of fazzy numbers:(9.217, 12.427, 15.637)
Difference of fazzy numbers: (2.337, 5.547, 8.757)
Composition of fazzy numbers: (5.88363, 30.9153, 58.4769)
Quotient of fazzy numbers: (1.37754, 2.6125, 13.6913)
Opposite fazzy number: (0.105854, 0.111272, 0.117275)

Sum of fazzy numbers:(83.1, 91.423, 99.746)
Difference of fazzy numbers: (-3.146, 5.177, 13.5)
Composition of fazzy numbers: (1695.6, 2082.84, 2487.18)
Quotient of fazzy numbers: (0.937388, 1.12005, 1.375)
Opposite fazzy number: (0.020202, 0.0207039, 0.0212314)

Sum of fazzy numbers:(13.36, 14.34, 15.32)
Difference of fazzy numbers: (11.46, 12.44, 13.42)
Composition of fazzy numbers: (8.862, 12.7205, 16.944)
Quotient of fazzy numbers: (10.55, 14.0947, 20.1714)
Opposite fazzy number: (0.0708215, 0.0746826, 0.0789889)

Sum of fazzy numbers:(7.17, 7.38, 7.59)
Difference of fazzy numbers: (1.61, 1.82, 2.03)
Composition of fazzy numbers: (12.188, 12.788, 13.392)
Quotient of fazzy numbers: (1.57706, 1.65468, 1.73285)
Opposite fazzy number: (0.208333, 0.217391, 0.227273)

Sum of fazzy numbers:(47.23, 49.26, 51.29)
Difference of fazzy numbers: (-47.67, -45.64, -43.61)
Composition of fazzy numbers: (80.901, 85.8845, 90.988)
Quotient of fazzy numbers: (0.035996, 0.0381454, 0.040484)
Opposite fazzy number: (0.543478, 0.552486, 0.561798)
First number < than second one
```

```
Sum of fazzy numbers:(62.405, 64.616, 66.827)
Difference of fazzy numbers: (49.073, 51.284, 53.495)
Composition of fazzy numbers: (315.802, 386.295, 459.232)
Quotient of fazzy numbers: (7.31002, 8.69337, 10.6301)
Opposite fazzy number: (0.0169348, 0.0172563, 0.0175902)

Sum of fazzy numbers:(33.247, 43.027, 52.807)
Difference of fazzy numbers: (-33.687, -23.907, -14.127)
Composition of fazzy numbers: (156.362, 319.945, 529.352)
Quotient of fazzy numbers: (0.144066, 0.285655, 0.487725)
Opposite fazzy number: (0.0743494, 0.104603, 0.176367)
First number < than second one

Sum of fazzy numbers:(119.803, 130.503, 141.203)
Difference of fazzy numbers: (87.837, 98.537, 109.237)
Composition of fazzy numbers: (708.955, 1830.37, 2970.49)
Quotient of fazzy numbers: (4.41473, 7.16511, 18.4975)
Opposite fazzy number: (0.00865951, 0.0087321, 0.00880592)

(4.44445, 4.13, 2.17)
```

Вывод:

При выполнении лабораторной работы был реализован пользовательский литерал для работы с константами созданного класса FazzyNumber. Пользовательские литералы помогают сделать код более читабельным, однако использовать их стоит только тогда, когда это действительно необходимо.

Исходный код:

FazzyNumber.h:

```
#ifndef FAZZYNUMBER_H
#define FAZZYNUMBER_H

#include <iostream>

class FazzyNumber
{
public:
    FazzyNumber operator+(const FazzyNumber& N);
    FazzyNumber operator-(const FazzyNumber& N);
    FazzyNumber operator*(const FazzyNumber& N);
    FazzyNumber operator/(const FazzyNumber& N);

    bool operator<(const FazzyNumber& N);
    bool operator>(const FazzyNumber& N);
    bool operator==(const FazzyNumber& N);
    bool operator<=(const FazzyNumber& N);
    bool operator>=(const FazzyNumber& N);
    bool operator!=(const FazzyNumber& N);

    friend std::ostream& operator<<(std::ostream& os, FazzyNumber other);
    friend std::istream& operator>>(std::istream& is, FazzyNumber other);

    FazzyNumber get_opposite();

    FazzyNumber(float l, float x, float r): l(l), x(x), r(r) {}
    FazzyNumber(): l(0), x(0), r(0) {}

private:
    float l, x, r;
};

FazzyNumber operator""_fn(const char* string, size_t size);

#endif
```

FazzyNumber.cpp:

```
#include "FazzyNumber.h"

FazzyNumber FazzyNumber::operator+(const FazzyNumber& N)
{
    return FazzyNumber(this->l + N.l, this->x + N.x, this->r + N.r);
}

FazzyNumber FazzyNumber::operator-(const FazzyNumber& N)
{
    return FazzyNumber(this->l - N.r, this->x - N.x, this->r - N.l);
}

FazzyNumber FazzyNumber::operator*(const FazzyNumber& N)
{
    return FazzyNumber(this->l * N.l, this->x * N.x, this->r * N.r);
}

FazzyNumber FazzyNumber::operator/(const FazzyNumber& N)
{
    if (N.x <= 0) {
        throw std::invalid_argument("Operation A/B, where B < 0\n");
    }
    return FazzyNumber(this->l / N.r, this->x / N.x, this->r / N.l);
}

FazzyNumber FazzyNumber::get_opposite()
{
    if (this->x <= 0) {
        throw std::invalid_argument("Operation of getting opposite of A, where A <
0\n");
    }
    return FazzyNumber(1 / this->r, 1 / this->x, 1 / this->l);
}

bool FazzyNumber::operator<(const FazzyNumber& N)
{
    return this->x < N.x;
}

bool FazzyNumber::operator>(const FazzyNumber& N)
{
    return this->x > N.x;
}
```

```
bool FazzyNumber::operator==(const FazzyNumber& N)
{
    return this->x == N.x;
}

bool FazzyNumber::operator<=(const FazzyNumber& N)
{
    return this->x <= N.x;
}

bool FazzyNumber::operator>=(const FazzyNumber& N)
{
    return this->x >= N.x;
}

bool FazzyNumber::operator!=(const FazzyNumber& N)
{
    return this->x != N.x;
}

std::ostream& operator<<(std::ostream& os, FazzyNumber other)
{
    os << "(" << other.l << ", " << other.x << ", " << other.r << ")";
    return os;
}

std::istream& operator>>(std::istream& is, FazzyNumber other)
{
    is >> other.l >> other.x >> other.r;
    return is;
}

FazzyNumber operator""_fn(const char* string, size_t size)
{
    std::string a = "";
    int ind = 0;
    float nums[3];
    for (int i = 0; i < 3; i++) {
        while(string[ind] != '|') {
            a += string[ind];
            ++ind;
        }
        nums[i] = atof(a.c_str());
        a = "";
        ++ind;
    }
    return FazzyNumber(nums[0], nums[1], nums[2]);
}
```

main.cpp:

```
#include "FazzyNumber.h"
#include <fstream>

int main()
{
    std::ifstream file("test_01.txt");
    int size;
    file >> size;
    for(int i = 0; i < size; i++) {
        float a, b, c, e, f, g;
        file >> a >> b >> c >> e >> f >> g;
        FazzyNumber N1(a, b, c); FazzyNumber N2(e, f, g);
        std::cout << "Sum of fazzy numbers:" << N1 + N2 << "\n";
        std::cout << "Difference of fazzy numbers: " << N1 - N2 << "\n";
        std::cout << "Composition of fazzy numbers: " << N1 * N2 << "\n";
        std::cout << "Quotient of fazzy numbers: " << N1 / N2 << "\n";
        std::cout << "Opposite fazzy number: " << N1.get_opposite() << "\n";
        if (N1 < N2) {
            std::cout << "First number < than second one\n";
        }
        if (N1 == N2) {
            std::cout << "First number is equal to second one\n";
        }
        std::cout << "\n";
    }
    FazzyNumber N = "4.444445|4.13|2.17"_fn;
    std::cout << N << "\n";
    return 0;
}
```