**1.**

//Пищик Е.С. М8О-206Б-19 Лабораторная работа №1 вариант 20

/\*Создать класс Bottle для работы с емкостями. Класс должен состоять из двух вещественных чисел: a - объем емкости в литрах и b – процент наполнения емкости (0 – пустая, 1 – полная). Реализовать операции сложения и вычитания, а также сравнения объектов класса бутылка. При сложении должен складываться фактический объем заполнения бутылок.\*/

#include <iostream>

#include <fstream>

#include <string>

class Bottle

{

//2 приватных поля, a\_ - объем емкости, b\_ - процент заполнения емкости

//a\_ >= 0, 0 <= b\_ <= 1

double a\_;

double b\_;

public:

// Конструктор класса

explicit Bottle(double const& a = 0, double const& b = 0) : a\_(a), b\_(b) {}

//Геттеры

double getA() { return a\_; }

double getB() { return b\_; }

};

//оператор сложения двух бутылок

Bottle operator+(Bottle & lhs, Bottle & rhs)

{

return Bottle(lhs.getA() + rhs.getA(), (lhs.getA() \* lhs.getB() + rhs.getA() \* rhs.getB()) / (lhs.getA() + rhs.getA()));

}

//оператор вычитания двух бутылок

Bottle operator-(Bottle & lhs, Bottle & rhs)

{

//разница объемов жидкости

double v\_filled = lhs.getA() \* lhs.getB() - rhs.getA() \* rhs.getB();

//разница максимальных объемов

double v = lhs.getA() - rhs.getA();

return Bottle(v > 0 ? v : -v, v\_filled > 0 ? v\_filled : -v\_filled);

}

//операторы сравнения

//сравнение по текущему объему жидкости

bool operator<(Bottle & lhs, Bottle & rhs) { return lhs.getA() \* lhs.getB() < rhs.getA() \* rhs.getB(); }

bool operator>(Bottle & lhs, Bottle & rhs) { return rhs < lhs; }

bool operator<=(Bottle & lhs, Bottle & rhs) { return !(rhs < lhs); }

bool operator>=(Bottle & lhs, Bottle & rhs) { return !(lhs < rhs); }

bool operator==(Bottle & lhs, Bottle & rhs) { return !(rhs < lhs) && !(lhs < rhs); }

bool operator!=(Bottle & lhs, Bottle & rhs) { return !(lhs == rhs); }

//Тестирующая функция, в файл с тестом должно быть записано 6 числе типа double

void TestFunction(double p\_11, double p\_12, double p\_21, double p\_22, double p\_31, double p\_32)

{

Bottle s1(p\_11, p\_12);

Bottle s2(p\_21, p\_22);

Bottle s3(p\_31, p\_32);

Bottle s;

Bottle t;

s = s1 + s2;

t = s2 - s1;

std::cout << std::endl <<"First bottle: a = " << s1.getA() << " b = " << s1.getB() << std::endl;

std::cout << "Second bottle: a = " << s2.getA() << " b = " << s2.getB() << std::endl;

std::cout << "Third bottle: a = " << s3.getA() << " b = " << s3.getB() << std::endl;

std::cout << "Sum first and second bottles: a = " << s.getA() << " b = " << s.getB() << std::endl;

std::cout << "Sub first and second bottles: a = " << t.getA() << " b = " << t.getB() << std::endl;

if (s1 < s2) std::cout << "first bottle < second bottle" << std::endl;

else std::cout << "first bottle > second bottle" << std::endl;

if (s1 == s3) std::cout << "first bottle = third bottle" << std::endl;

else std::cout << "first bottle != third bottle" << std::endl;

if (s1 == s2) std::cout << "first bottle = second bottle" << std::endl;

else std::cout << "first bottle != second bottle" << std::endl;

std::cout << "Completed\n" << std::endl;

}

int main()

{

std::string name = "start";

std::cout << "Enter q or Q or quit or Quit for close programm" << std::endl;

//Бесконечный дикл до ввода ключевого слова для выхода

while(true)

{

double a = 0, b = 0, c = 0, d = 0, e = 0, f = 0;

std::cout << "Enter test file name: ";

std::cin >> name;

if (name == "Q" || name == "q" || name == "Quit" || name == "quit") break;

//поток ввода из файла, имя файла хранится в string переменной name

std::ifstream finp(name);

//взятие из файла 6 чисел и запись в переменные

finp >> a >> b >> c >> d >> e >> f;

TestFunction(a, b, c, d, e, f);

finp.close();

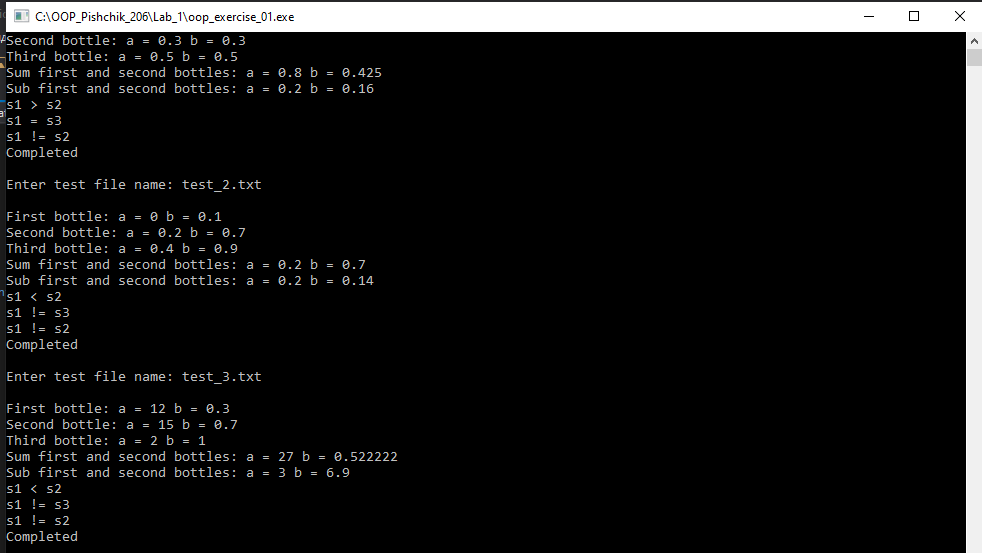
}

return 0;

}

**2.** <https://github.com/Pe4enIks/OOP_Pishchik_206>

**3-4.**

****

**5.** В программе создается 3 объекта типа Bottle и инициализируются значениями из файла. На экран выводятся поля каждого объекта Bottle, также суммы, разности 1 и 2 объекта Bottle, и при помощи операторов сравнений на экран выводится отношения между объектами класса Bottle.