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: *Попов Андрей Викторович, 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 №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.

**Описание программы:**

Исходный код разделён на 3 файла:

* bottle.h – описание класса адрес
* bottle.cpp – реализация класса адрес
* main.cpp – основная программа

**Дневник отладки:**

Проблем не возникло.

**Вывод:**  
 В рамках данной лабораторной работы я поработал с важной вещью - перегрузкой операторов. Также я познакомился с пользовательскими литералами. Это очень удобная и практическая вещь, о которой я не знал до изучения курса ООП. Использование этого средства позволяет получать из заданных типов данных какие-либо другие данные.

**Исходный код:**

**bottle.h**

**#ifndef BOTTLE\_H**

**#define BOTTLE\_H**

**#include <iostream>**

**class Bottle {**

**public:**

**Bottle();**

**Bottle(double a, double b);**

**Bottle(std::istream &is);**

**void Difference(const Bottle &c);**

**void Summa(const Bottle &c);**

**void Compare\_volume(const Bottle &c);**

**void Compare\_filling(const Bottle &c);**

**void Print(std::ostream &os);**

**Bottle operator + (const Bottle &object);**

**Bottle operator - (const Bottle &object);**

**bool operator == (const Bottle &object);**

**bool operator > (const Bottle &other);**

**bool operator < (const Bottle &other);**

**friend std::istream& operator >> (std::istream& is, Bottle &object);**

**friend std::ostream& operator << (std::ostream& os, Bottle &object);**

**~Bottle();**

**private:**

**double volume;**

**double filling;**

**};**

**#endif**

**bottle.cpp**

**#include "bottle.h"**

**#include <cmath>**

**Bottle::Bottle() {**

**volume = 0.0;**

**filling = 0.0;**

**std:: cout << "Start: " << std:: endl;**

**}**

**Bottle::Bottle(double a, double b) {**

**if (a > 0.0 && b > 0.0 && b > 1){**

**volume = a;**

**filling = b;**

**}**

**else {**

**std:: cout << "Please enter correct numbers!" << std:: endl;**

**}**

**std:: cout << "The bottle according to your parameters has been created" << std:: endl;**

**}**

**Bottle::Bottle(std::istream &is) {**

**std:: cout << "Please enter your bottle data: " << std:: endl;**

**is >> volume >> filling;**

**if ((volume <= 0.0) || (filling <= 0.0) || (filling > 1)) {**

**std:: cout << "Invalind input. Enter again!" << std:: endl;**

**is >> volume >> filling;**

**}**

**std:: cout << "The bottle has been created via istream" << std:: endl;**

**}**

**void Bottle::Difference(const Bottle &c) {**

**double differ = volume\*filling - c.volume\*c.filling;**

**std:: cout << "Difference: " << differ << std:: endl;**

**}**

**void Bottle::Summa(const Bottle &c) {**

**double bottle = volume\*filling + c.volume\*c.filling;**

**std:: cout << "Total volume: " << bottle << std:: endl;**

**}**

**void Bottle::Compare\_volume(const Bottle &c) {**

**if (volume > c.volume){**

**std:: cout << "The first bottle is larger!" << std:: endl;**

**}**

**else if (volume == c.volume) {**

**std:: cout << "Volumes are equal!" << std:: endl;**

**}**

**else {**

**std:: cout << "The second bottle is larger!" << std:: endl;**

**}**

**}**

**void Bottle::Compare\_filling(const Bottle &c) {**

**if (volume\*filling > c.volume\*c.filling){**

**std:: cout << "The filling of first bottle is larger!" << std:: endl;**

**}**

**else if (volume\*filling == c.volume\*c.filling) {**

**std:: cout << "Bottles are equal!" << std:: endl;**

**}**

**else {**

**std:: cout << "The filling of second bottle is larger!" << std:: endl;**

**}**

**}**

**void Bottle::Print(std::ostream& os) {**

**os << "Your bottle is: " << volume << ", " << filling << ", " << volume\*filling << "l" << std:: endl;**

**}**

**std::istream& operator >> (std::istream& is, Bottle &object) {**

**std:: cout << "Please enter your bottle data: " << std:: endl;**

**is >> object.volume >> object.filling;**

**if ((object.volume <= 0.0) || (object.filling <= 0.0) || (object.filling > 1)) {**

**std:: cout << "Invalind input. Enter again!" << std:: endl;**

**is >> object.volume >> object.filling;**

**}**

**std:: cout << "The bottle has been created via istream" << std:: endl;**

**return is;**

**}**

**Bottle Bottle::operator + (const Bottle &other) {**

**this->volume += other.volume;**

**this->filling = 1;**

**return \*this;**

**}**

**Bottle Bottle::operator - (const Bottle &other) {**

**this->volume -= other.volume;**

**this->filling = 1;**

**return \*this;**

**}**

**bool Bottle::operator < (const Bottle &other) {**

**if (volume\*filling < other.volume\*other.filling) {**

**return true;**

**}**

**else {**

**return false;**

**}**

**}**

**bool Bottle::operator > (const Bottle &other) {**

**if (volume\*filling > other.volume\*other.filling) {**

**return true;**

**}**

**else {**

**return false;**

**}**

**}**

**std::ostream& operator << (std::ostream& os, Bottle &object) {**

**os << object.volume << ";" << object.filling<< std:: endl;**

**return os;**

**}**

**bool Bottle::operator == (const Bottle &object) {**

**if (volume == object.volume && filling == object.filling) {**

**return true;**

**}**

**return false;**

**}**

**Bottle::~Bottle() {**

**std:: cout << "FROM DESTRUCTOR: Your bottle has been deleted" << std:: endl;**

**}**

**main.cpp**

**#include "bottle.h"**

**unsigned long long operator "" \_tom3(unsigned long long littres) {**

**unsigned long long m3 = littres / 1000;**

**return m3;**

**}**

**int main(){**

**Bottle a;**

**std:: cin >> a;**

**Bottle b;**

**std:: cin >> b;**

**Bottle c = a + b;**

**std:: cout << c;**

**Bottle d;**

**std:: cin >> d;**

**Bottle f;**

**std:: cin >> f;**

**Bottle x = d - f;**

**std:: cout << x;**

**return 0;**

**}**

**Пример работы:**

**Start:**

**Please enter your bottle data:**

**200**

**0.5**

**The bottle has been created via istream**

**Start:**

**Please enter your bottle data:**

**100**

**0.8**

**The bottle has been created via istream**

**300;1**

**Start:**

**Please enter your bottle data:**

**80**

**1**

**The bottle has been created via istream**

**Start:**

**Please enter your bottle data:**

**50**

**0.7**

**The bottle has been created via istream**

**30;1**

**FROM DESTRUCTOR: Your bottle has been deleted**

**FROM DESTRUCTOR: Your bottle has been deleted**

**FROM DESTRUCTOR: Your bottle has been deleted**

**FROM DESTRUCTOR: Your bottle has been deleted**

**FROM DESTRUCTOR: Your bottle has been deleted**

**FROM DESTRUCTOR: Your bottle has been deleted**