**Московский авиационный институт (национальный исследовательский университет)**

**Институт информационных технологий и прикладной математики**

**Кафедра вычислительной математики и программирования**

**Лабораторная работа №1 по курсу «Объектно-ориентированное программирование»**

Студент: И. С. Глушатов

Преподаватель: Л. Н. Чернышов

Группа: М8О-207Б-19

Дата:

Оценка:

Подпись:

**Москва, 2020**

**Лабораторная работа №1**

**Задача:** Изучение системы сборки на языке C++, изучение систем контроля версии. Изучение основ работы с классами в С++;

**Вариант:** Создать класс Position для работы с географическими координатами. Координаты задаются двумя числами широта и долгота. Долгота находится в диапазоне от -180 до 180 градусов. Широта находится в диапазоне от -90 до 90 градусов. Реализовать арифметические операции сложения, вычитания, умножения и деления, а также операции сравнения.

В классе Position я создаю два поля: latitude и longitude (широта и долгота). Так как обе эти величины находятся в определенном диапазоне я создал две функции, которые выравнивают их, если значения выходят за пределы: AlignLatitude и AlignLongitude.

Далее переопределяю операторы сложения, вычитания, умножения и деления, при этом выравнивая полученные значения, а так же операторы сравнения.

В файле main.cpp на вход подается четыре переменных целого типа, как координаты для двух переменных типа Position и еще два числа (на которое надо умножить и на которое надо разделить). После этих операций выдается результат сравнения переменных a и b.

Если пользователь попытается разделить на 0, то переменная выдаст сообщение об ошибке операции и переменная останется того же значения.

**Код программы**

**main.cpp**

#include <iostream>

#include <typeinfo>

#include "Position.hpp"

int main() {

Position a; Position b; int num;

std::cout << "Введите a (два целых числа через пробел): "; std::cin >> a; std::cout << "\n";

std::cout << "Введите b (два целых числа через пробел): "; std::cin >> b; std::cout << "\n";

std::cout << "a = " << a << "\n" << "b = " << b << "\n";

std::cout << "Сумма a и b: " << a+b << "\n";

std::cout << "Разность a и b: " << a-b << "\n";

std::cin.clear();

std::cin.ignore(32767,'\n');

std::cout << "Введите число, на которое хотите умножить переменную а: "; std::cin >> num; std::cout << "\n";

std::cout << "Умножение a на число " << num << ": " << a\*num << "\n";

std::cin.clear();

std::cin.ignore(32767,'\n');

std::cout << "Введите число на которое хотите разделить переменную а: "; std::cin >> num; std::cout << "\n";

std::cout << "Деление a на число " << num << ": " << a/num << "\n";

std::cout << "Меньше: " << (a<b) << "\n";

std::cout << "Больше: " << (a>b) << "\n";

std::cout << "Равно: " << (a==b) << "\n";

std::cout << "Меньше или равно: " << (a<=b) << "\n";

std::cout << "Больше или равно: " << (a>=b) << "\n\n";

return 0;

}

**Position.hpp**

#ifndef POSITION\_HPP

#define POSITION\_HPP

struct Position {

int latitude; // широта в диапозоне от -180 до 180

int longitude; // долгота в диапозоне от -90 до 90

Position(): latitude(0), longitude(0) {};

Position(int lat, int lon);

~Position(){};

Position operator+(Position rval);

Position operator-(Position rval);

Position operator\*(int rval);

Position operator/(int rval);

bool operator<=(Position rval);

bool operator>=(Position rval);

bool operator<(Position rval);

bool operator>(Position rval);

bool operator==(Position rval);

};

int AlignLatitude(int lat) {

lat%=360;

if (lat%360>=-180 and lat%360<=180) return lat;

if (lat>180) return lat - 360;

if (lat<180) return lat + 360;

return 0;

}

int AlignLongitude(int lon) {

lon%=360;

if (lon>=-90 and lon<=90) return lon;

if (lon>90 and lon<=270) return 180-lon;

if (lon>270 and lon<360) return lon-360;

if (lon>=-360 and lon<=-270) return 360+lon;

if (lon>-270 and lon<-90) return -(lon+180);

return 0;

}

Position::Position(int lat, int lon): latitude(AlignLatitude(lat)), longitude(AlignLongitude(lon)) {}

Position Position::operator+(Position rval) {

return Position(AlignLatitude(latitude+rval.latitude), AlignLongitude(longitude+rval.longitude));

}

Position Position::operator-(Position rval) {

return Position(AlignLatitude(latitude-rval.latitude), AlignLongitude(longitude-rval.longitude));

}

Position Position::operator\*(int rval) {

return Position(AlignLatitude(latitude\*rval), AlignLongitude(longitude\*rval));

}

Position Position::operator/(int rval) {

if (!rval) {

std::cout << "Деление на ноль! ";

return Position(AlignLatitude(latitude), AlignLongitude(longitude));

}

return Position(AlignLatitude(latitude/rval), AlignLongitude(longitude/rval));

}

std::istream& operator>>(std::istream& in, Position& rval) {

int a, b;

while (1) {

in >> a >> b;

if (std::cin.fail()) {

std::cin.clear();

std::cin.ignore(32767,'\n');

std::cout << "Неправильный ввод. Попробуйте снова: ";

} else {

break;

}

}

std::cout << a << b <<std::endl;

rval.latitude = AlignLatitude(a);

rval.longitude = AlignLongitude(b);

return in;

}

bool Position::operator<(Position rval) {

return (latitude<rval.latitude) && (longitude<rval.longitude);

}

bool Position::operator>(Position rval) {

return ((latitude>rval.latitude) && (longitude>rval.longitude));

}

bool Position::operator<=(Position rval) {

return ((latitude<rval.latitude) && (longitude<rval.longitude)) || ((latitude==rval.latitude) && (longitude==rval.longitude));

}

bool Position::operator>=(Position rval) {

return ((latitude>rval.latitude) && (longitude>rval.longitude)) || ((latitude==rval.latitude) && (longitude==rval.longitude));

}

bool Position::operator==(Position rval) {

return (latitude==rval.latitude) && (longitude==rval.longitude);

}

std::ostream& operator<<(std::ostream& out, const Position& rval) {

return out << rval.latitude << " " << rval.longitude;

}

#endif

**Makefile**

CC = g++

FLAGS = -std=c++11 -pedantic -Wall -Werror -Wextra

all:

${CC} ${FLAGS} main.cpp -o oop\_exercise\_01

run:

./oop\_exercise\_01 < test0.txt

./oop\_exercise\_01 < test1.txt

./oop\_exercise\_01 < test2.txt

./oop\_exercise\_01 < test3.txt

./oop\_exercise\_01 < test4.txt

**Ссылка на github**

https://github.com/Igor743646/OOP/tree/master/oop\_exercise\_01

**Тесты**

**Тест №1:**

4 5

6 4

3

4

**Тест №2:**

45 53

-75 -91

8

0

**Тест №3:**

435 2452

25 -24346

23

7

**Тест №4:**

0 0

-2 0

9

-1

**Тест №5:**

756 dfvdf

vsdv 43vv

43 55

vsv 35

43 415

4

9

По описанию реализации программы мы создаем несколько тестов, где вводим дважды по 2 числа для переменных a и b, после чего вводим еще два для проверки умножения и деления. Если пользователь введет неправильные данные для аргументов класса Position, то их ввод будет запрошен заново.

**Консоль**

igor@igor-Aspire-A315-53G:~/Рабочий стол/c++/OOP/lab1$ make run

./oop\_exercise\_01 < test0.txt

Введите a (два целых числа через пробел): 4 5

Введите b (два целых числа через пробел): 6 4

a = 4 5

b = 6 4

Сумма a и b: 10 9

Разность a и b: -2 1

Введите число, на которое хотите умножить переменную а:

Умножение a на число 3: 12 15

Введите число на которое хотите разделить переменную а:

Деление a на число 4: 1 1

Меньше: 0

Больше: 0

Равно: 0

Меньше или равно: 0

Больше или равно: 0

./oop\_exercise\_01 < test1.txt

Введите a (два целых числа через пробел): 45 53

Введите b (два целых числа через пробел): -75 -91

a = 45 53

b = -75 -89

Сумма a и b: -30 -36

Разность a и b: 120 38

Введите число, на которое хотите умножить переменную а:

Умножение a на число 8: 0 64

Введите число на которое хотите разделить переменную а:

Деление a на число 0: Деление на ноль! 45 53

Меньше: 0

Больше: 1

Равно: 0

Меньше или равно: 0

Больше или равно: 1

./oop\_exercise\_01 < test2.txt

Введите a (два целых числа через пробел): 435 2452

Введите b (два целых числа через пробел): 25 -24346

a = 75 -68

b = 25 46

Сумма a и b: 100 -22

Разность a и b: 50 -66

Введите число, на которое хотите умножить переменную а:

Умножение a на число 23: -75 -56

Введите число на которое хотите разделить переменную а:

Деление a на число 7: 10 -9

Меньше: 0

Больше: 0

Равно: 0

Меньше или равно: 0

Больше или равно: 0

./oop\_exercise\_01 < test3.txt

Введите a (два целых числа через пробел): 0 0

Введите b (два целых числа через пробел): -2 0

a = 0 0

b = -2 0

Сумма a и b: -2 0

Разность a и b: 2 0

Введите число, на которое хотите умножить переменную а:

Умножение a на число 9: 0 0

Введите число на которое хотите разделить переменную а:

Деление a на число -1: 0 0

Меньше: 0

Больше: 0

Равно: 0

Меньше или равно: 0

Больше или равно: 0

./oop\_exercise\_01 < test4.txt

Введите a (два целых числа через пробел): Неправильный ввод. Попробуйте снова: Неправильный ввод. Попробуйте снова: 43 55

Введите b (два целых числа через пробел): Неправильный ввод. Попробуйте снова: 43 415

a = 43 55

b = 43 55

Сумма a и b: 86 70

Разность a и b: 0 0

Введите число, на которое хотите умножить переменную а:

Умножение a на число 4: 172 -40

Введите число на которое хотите разделить переменную а:

Деление a на число 9: 4 6

Меньше: 0

Больше: 0

Равно: 1

Меньше или равно: 1

Больше или равно: 1

igor@igor-Aspire-A315-53G:~/Рабочий стол/c++/OOP/lab1

**Выводы**

В ходе лабораторной работы я научился создавать классы и объекты в языке С++, переопределять операторы для этого класса, в том числе и операторы ввода и вывода.

Собрал проект с помощью makefile. Познакомился с github и залил туда свой проект.

**Список литературы**

* Поисковик - Google.

URL: [https://www.google.com/](http://www.google.com/)

* Классы, объекты и методы C++

URL: https://ravesli.com/urok-113-klassy-obekty-i-metody-klassov/

* Про операторы C++ и их переопределение

URL: https://en.cppreference.com/w/cpp/language/operators