**Московский авиационный институт**

**(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Объектно-ориентированное программирование»

**Лабораторная работа № 2**

Тема: Перегрузка операторов в С++

Студент: Воронов Кирилл Михайлович

Группа: 80-207

Преподаватель: Чернышов Л.Н.

Дата:

Оценка:

Москва, 2020

1. **Постановка задачи**

Изучение механизмов перегрузки операторов и механизмов работы пользовательскими литералами.

***Вариант 16***

**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 виде перегрузки операторов. Необходимо реализовать пользовательский литерал для работы с константами типа **Position**.

1. **Описание программы**

Для хранения координат создадим у класса **Position** две переменные: dol для хранения долготы и shir для хранения широты. Также реализуем перегрузку операторов сложения, вычитания, умножения и деления на число, сравнения и вывода. Создадим пользовательский литерал для работы с константами типа **Position** и два конструктора: с параметрами и без. Взаимодействие с программой происходит через терминал, пользователь вводит координаты, после чего выполняются соответствующие операции.

1. **Набор тестов**

Тест 1

a

Тест 2

1 2

3 4

2

1

Тест 3

48 53

180 56

7

2

Тест 4

69.2 21.66

41.4 12.5

2

0

Тест 5

179.9 89.89

23.2 56.475

9

29

1. **Результаты выполнения тестов**

Тест 1

Пользовательские литералы

44 42

-124 -42

-12.4 -4.2

Введите две координаты

a

Вы ввели неверные координаты

Тест 2

Пользовательские литералы

44 42

-124 -42

-12.4 -4.2

Введите две координаты

1 2

3 4

Сумма

4 6

Разность

-2 -2

Умножение

Введите число

2

2 4

6 8

Деление

Введите число

1

1 2

3 4

Сравнение

Второе число больше

Тест 3

Пользовательские литералы

44 42

-124 -42

-12.4 -4.2

Введите две координаты

48 53

180 56

Сумма

48 19

Разность

-132 -42

Умножение

Введите число

7

156 11

0 32

Деление

Введите число

2

24 26.5

90 28

Сравнение

Второе число больше

Тест 4

Пользовательские литералы

44 42

-124 -42

-12.4 -4.2

Введите две координаты

69.2 21.66

41.4 12.5

Сумма

110.6 34.16

Разность

27.8 27.8

Умножение

Введите число

2

138.4 43.32

82.8 25

Деление

Введите число

0

Деление невозможно

Сравнение

Первое число больше

Тест 5

Пользовательские литералы

44 42

-124 -42

-12.4 -4.2

Введите две координаты

179.9 89.89

23.2 56.475

Сумма

23.1 56.365

Разность

156.7 66.7

Умножение

Введите число

9

179.1 89.01

28.8 58.275

Деление

Введите число

29

6.20345 3.09966

0.8 1.94741

Сравнение

Первое число больше

1. **Листинг программы**

/\* Воронов К.М. М8О-207Б-19

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 виде перегрузки операторов.

Необходимо реализовать пользовательский литерал для работы с константами типа Position.

\*/

#include <iostream>

using namespace std;

class Position{

public:

double dol; //-180 180

double shir; //-90 90

Position(): dol(0),shir(0) {};

Position(double a, double b){

dol=a;

shir=b;

}

friend Position operator+(const Position &a, const Position &b);

friend Position operator-(const Position &a, const Position &b);

friend Position operator\*(const Position &a, const Position &b);

friend Position operator/(const Position &a, const Position &b);

friend bool operator>(const Position &a, const Position &b);

friend bool operator<(const Position &a, const Position &b);

friend bool operator==(const Position &a, const Position &b);

friend ostream & operator << (std::ostream &out, const Position &a);

~Position(){};

};

double mod(double a,int n){

int c=a/n;

return a-c\*n;

}

/\* Реализация операторов класса \*/

Position operator+(const Position &a, const Position &b){

return Position (mod((a.dol+b.dol),180),mod((a.shir+b.shir),90));

}

Position operator-(const Position &a, const Position &b){

return Position (mod((a.dol-b.dol),180),mod((a.dol-b.dol),90));

}

Position operator\*(const Position &a, int ch){

return Position (mod((a.dol\*ch),180),mod((a.shir\*ch),90));

}

Position operator/(const Position &a, int ch){

return Position (mod((a.dol/ch),180),mod((a.shir/ch),90));

}

bool operator>(const Position &a, const Position &b){

if(a.dol!=b.dol)

return a.dol>b.dol;

else

return a.shir>b.shir;

}

bool operator<(const Position &a, const Position &b){

if(a.dol!=b.dol)

return a.dol<b.dol;

else

return a.shir<b.shir;

}

bool operator==(const Position &a, const Position &b){

if((a.dol==b.dol)&&(a.shir==b.shir))

return true;

else

return false;

}

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

out << a.dol <<" "<< a.shir ;

return out;

}

/\* Реализация парсера для литерала \*/

Position operator "" \_position(const char\* st, size\_t n){

double dol=0,shir=0;

string s(st);

int l=0, u=0;

for(int i = 0; i < n; ++i){

if(s[i]==','){

if(u==0){

dol=stod(s.substr(l,i-l+1));

}

l=i+1;

break;

}

}

shir=stod(s.substr(l,n));

return Position(dol,shir);

}

int main(){

printf("Пользовательские литералы\n");

cout<<"44,42"\_position<<endl;

cout<<"-124,-42"\_position<<endl;

cout<<"-12.4,-4.2"\_position<<endl;

Position a,b,c;

double ch;

printf("Введите две координаты\n");

cin >> a.dol >> a.shir >> b.dol >> b.shir;

if((!cin) || (a.dol>180) || (a.dol<-180) || (a.shir>90)||(a.shir<-90)|| (b.dol>180) || (b.dol<-180) || (b.shir>90)||(b.shir<-90)){

printf("Вы ввели неверные координаты\n");

return 1;

}

printf("Сумма\n");

c = a + b;

cout<<c.dol<<" "<<c.shir<<endl;

printf("Разность\n");

c = a - b;

cout<<c.dol<<" "<<c.shir<<endl;

printf("Умножение\n");

printf("Введите число\n");

cin>>ch;

if(!cin){

printf("Вы ввели неверное число\n");

}else{

c = a \* ch;

cout<<c.dol<<" "<<c.shir<<endl;

c = b \* ch;

cout<<c.dol<<" "<<c.shir<<endl;

}

printf("Деление\n");

printf("Введите число\n");

cin>>ch;

if(!cin){

printf("Вы ввели неверное число\n");

}

else{

if (ch==0)

printf("Деление невозможно\n");

else{

c = a / ch;

cout<<c.dol<<" "<<c.shir<<endl;

c= b / ch;

cout<<c.dol<<" "<<c.shir<<endl;

}

}

printf("Сравнение\n");

if(a==b)

printf("Числа равны\n");

else if(a>b)

printf("Первое число больше\n");

else printf("Второе число больше\n");

}

1. **Вывод**

Я научился перегружать операторы в C++ для различных операций, а также создавать и использовать пользовательские литералы.

**Список литературы**

1.Пользовательские литералы в С++11[Электронный ресурс]

URL:<https://habr.com/ru/post/140357/> (дата обращения 13.10.2020).

2.Перегрузка операторов С++[Электронный ресурс]

URL:<https://habr.com/ru/post/132014/> (дата обращения 13.10.2020).