**Московский авиационный институт**

**(Национальный исследовательский университет)**

Институт: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Объектно-ориентированное программирование»

**Лабораторная работа № 1**

Тема: Простые классы на языке С++

Студент: Хренникова Ангелина Сергеевна

Группа: 8О-208

Преподаватель: Чернышов Л.Н.

Дата: 17.09.2020

Оценка:

Москва, 2020

**Содержание**

1. Постановка задачи
2. Описание программы
3. Набор тестов
4. Результаты выполнения тестов
5. Листинг программы
6. Вывод
7. Список литературы

**Постановка задачи**

Цель:

* Изучить системы сборки на языке C++, изучить системы контроля версии;
* Изучить основы работы с классами в С++.

Задача: Разработать программу на языке C++ согласно варианту задания. Программа на C++ должна собираться с помощью системы сборки CMake. Программа должна получать данные из стандартного ввода и выводить данные в стандартный вывод.

Вариант(№10): Создать класс Angle для работы с углами на плоскости, задаваемыми величиной в градусах и минутах. Обязательно должны быть реализованы: перевод в радианы, приведение к диапазону 0–360, сложение и вычитание углов, деление углов, получение значений тригонометрических функций, сравнение углов.

**Описание программы**

Описана глобальная переменная, хранящая значение числа π, которое необходимо для перевода значения угла в радианы.

Класс Angle, определенный в программе, содержит три поля: degree - поле значения числа градусов, minute - поле значения числа минут, alpha - поле значения угла в минутах(т. е. degree \* 60 + minute).

В методах класса содержится конструктор для инициализации полей объекта, конструктор копирования, функция для считывания данных из стандартного потока ввода в объект класса, функция для возвращения промежуточного значения, функция для вывода в формате: градусы и минуты, в диапазоне 0-360, в радианах, а также перегруженные операторы сложения, вычитания, деления, перегруженные операторы сравнения.

Меню, с помощью которого пользователь может выбрать необходимое действие (перевод в радианы; приведение к диапазону 0-360; сумма, разность, частное углов; сравнение углов; вычисление тригонометрической функции). Для пункта, в котором необходимо произвести вычисление значения тригонометрической функции, создано еще одно меню, позволяющее выбрать необходимую тригонометрическую функцию.

Главная функция выводит меню, после чего пользователь должен выбрать один из пунктов. В каждом из пунктов создается объект или несколько объектов класса, производится запрос на ввод пользователем данных, после чего с помощью описанных функций производятся необходимые действия и выводится ответ. При неверном выборе пункта меню, выводится сообщение об ошибке. Если не был выбран пункт о выходе из программы, запрашивается повторный выбор пункта меню.

**Набор тестов**

Входными данными являются пункты меню и значения угла в градусах и минутах.

Входные данные теста 1: 1 13 0 2 543 12 3 14 56 7 22 4 100 34 1 14 5 78 22 39 11 6 78 52 47 10 7 30 0 1 7 45 0 3 8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Действие | Градусы1 | Минуты1 | Градусы2 | Минуты2 | Результат |
| 1 | В радианах | 13 | 0 | - | - | 0.2269 |
| 2 | В диапазоне 0-360 | 543 | 12 | - | - | 183° 12’ |
| 3 | Сумма | 14 | 56 | 7 | 22 | 22° 18’ |
| 4 | Разность | 100 | 34 | 1 | 14 | 99° 20’ |
| 5 | Частное | 78 | 22 | 39 | 11 | 2 |
| 6 | Сравнение | 78 | 52 | 47 | 10 | >=, >, != |
| 7 | Синус | 30 | 0 | - | - | 0.5 |
| 7 | Тангенс | 45 | 0 | - | - | 1 |
| 8 | Выход из программы | - | - | - | - | - |

Входные данные теста 2: 5 82 12 74 2 7 468 4 1 54 3 8

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Действие | Градусы1 | Минуты1 | Градусы2 | Минуты2 | Результат |
| 5 | Частное | 82 | 12 | 74 | 2 | 1,1103 |
| 7 | Котангенс | 468 | 1 | - | - | -0,3252 |
| 1 | В радианах | 54 | 3 | - | - | 0,9434 |
| 8 | Выход из программы | - | - | - | - | - |

**Результаты выполнения тестов**

Тест 1:

Menu:

1. Convert angle to radians.

2. Convert angle to range 0-360.

3. Fold two angles.

4. Subtract two angles.

5. Divide one angle by another.

6. Compare angles.

7. Calculate trigonometric function.

8. Exit.

Select menu item:

1

Enter degrees:13

Enter minutes:0

Angle in radians: 0.226893

Select menu item:

2

Enter degrees:543

Enter minutes:12

Angle to range 0-360: degrees: degree: 183, minute: 12.

Select menu item:

3

Enter degrees:14

Enter minutes:56

Enter degrees:7

Enter minutes:22

Sum of who angles: degree: 22, minute: 18.

Select menu item:

4

Enter degrees:100

Enter minutes:34

Enter degrees:1

Enter minutes:14

Difference of two angles: degree: 99, minute: 20.

Select menu item:

5

Enter degrees:78

Enter minutes:22

Enter degrees:39

Enter minutes:11

Result of dividing one angle by another: 2

Select menu item:

6

Enter degrees:78

Enter minutes:52

Enter degrees:47

Enter minutes:10

78.8667 >= 47.1667

78.8667 > 47.1667

78.8667 != 47.1667

Select menu item:

7

Enter degrees:30

Enter minutes:0

Trigonometric functions:

1. Sin.

2. Cos.

3. Tg.

4. Ctg.

Select the trigonometric function you want to calculate:

1

Sinus: 0.5

Select menu item:

7

Enter degrees:45

Enter minutes:0

Trigonometric functions:

1. Sin.

2. Cos.

3. Tg.

4. Ctg.

Select the trigonometric function you want to calculate:

3

Tangents: 1

Select menu item:

8

Тест 2:

Menu:

1. Convert angle to radians.

2. Convert angle to range 0-360.

3. Fold two angles.

4. Subtract two angles.

5. Divide one angle by another.

6. Compare angles.

7. Calculate trigonometric function.

8. Exit.

Select menu item:

5

Enter degrees:82

Enter minutes:12

Enter degrees:74

Enter minutes:2

Result of dividing one angle by another: 1.11031

Select menu item:

7

Enter degrees:468

Enter minutes:1

Trigonometric functions:

1. Sin.

2. Cos.

3. Tg.

4. Ctg.

Select the trigonometric function you want to calculate:

4

Cotangents: -0.325241

Select menu item:

1

Enter degrees:54

Enter minutes:3

Angle in radians: 0.94335

Select menu item:

8

**Листинг программы**

/\* Лабораторная работа №1

по Объектно - Ориентированному Программированию

Хренникова Ангелина, М80-208Б-19, 16.09.2020\*/

#include <iostream>

#include <cmath>

using namespace std;

const double PI = 3.141592653589793238463;

class Angle { // класс углов на плоскости, задаваемых градусами и минутами

private:

int degree; //градусы

int minute; //минуты

double alpha; //угол в минутах(градусы + минуты)

public:

Angle() : degree(0), minute(0), alpha(0.0 + (degree + minute / 60)) {} //конструктор

Angle(int d0, int m0, double a0) : degree(d0), minute(m0), alpha(a0) {} //конструктор копирования

void getangle() { // Приглашение для ввода значений

cout << "Enter degrees:";

cin >> degree;

cout << "Enter minutes:";

cin >> minute;

alpha = 0.0 + (degree \* 60 + minute);

}

double returning() {

return alpha; // промежуточное значение, для удобства расчетов

}

void result() {

int f = minute % 60;

int k = minute / 60;

cout << " degree: " << degree + k << ", minute: " << f << "." << endl; //вывод градусов и минут

}

void standart() {

int g = ((degree \* 60 + minute) % 21600) / 60;

int j = ((degree \* 60 + minute) % 21600) % 60;

cout << " degree: " << g << ", minute: " << j << "." << endl; //приведение к диапазону 0-360

}

double radian() {

return ((alpha \* 2 \* PI) / 21600); //перевод в радианы

}

// Перегрузка операторов

Angle& operator=(const Angle& other) {

degree = other.degree;

minute = other.minute;

alpha = other.alpha;

return \*this;

}

Angle operator+(const Angle& other) const {

return Angle(degree + other.degree, minute + other.minute, alpha + other.alpha);

}

Angle operator-(const Angle& other) const {

return Angle(degree - other.degree, minute - other.minute, alpha - other.alpha);

}

Angle operator/(const Angle& other) const {

return Angle(degree / other.degree, minute / other.minute, alpha / other.alpha);

}

bool operator<(const Angle& other) const {

return alpha < other.alpha;

}

bool operator>(const Angle& other) const {

return alpha > other.alpha;

}

bool operator!=(const Angle& other) const {

return alpha != other.alpha;

}

bool operator==(const Angle& other) const {

return alpha == other.alpha;

}

bool operator>=(const Angle& other) const {

return alpha >= other.alpha;

}

bool operator<=(const Angle& other) const {

return alpha <= other.alpha;

}

};

void menu() {

cout << "Menu:\n"; //Меню

cout << "1. Convert angle to radians.\n"; //Перевести угол в радианы

cout << "2. Convert angle to range 0-360.\n"; //Привести угол к диапазону 0-360

cout << "3. Fold two angles.\n"; //Сложить два угла

cout << "4. Subtract two angles.\n"; //Вычесть два угла

cout << "5. Divide one angle by another.\n"; //Разделить один угол на другой

cout << "6. Compare angles.\n"; //Сравнить углы

cout << "7. Calculate trigonometric function.\n"; //Вычислить тригонометрические функции

cout << "8. Exit.\n"; //Выйти из программы

}

void trig() {

cout << "Trigonometric functions:\n"; //Тригонометрические функции

cout << "1. Sin.\n"; //Синус

cout << "2. Cos.\n"; //Косинус

cout << "3. Tg.\n"; //Тангенс

cout << "4. Ctg.\n"; //Котангенс

}

int main() {

int op;

menu();

cout << "Select menu item:\n";

cin >> op;

while (op >= 1 && op <= 8) {

if (op == 1) { // Перевод угла в радианы

Angle ang1;

ang1.getangle();

cout << "Angle in radians: " << ang1.radian() << endl;

}

else if (op == 2) { // Приведение угла к диапазону 0-360

Angle ang2;

ang2.getangle();

cout << "Angle to range 0-360: degrees: ";

ang2.standart();

}

else if (op == 3) { // Сумма углов

Angle ang3, ang4, ang5;

ang3.getangle();

ang4.getangle();

ang5 = ang3 + ang4;

cout << "Sum of who angles: ";

ang5.result();

}

else if (op == 4) { // Разность углов

Angle ang6, ang7, ang8;

ang6.getangle();

ang7.getangle();

ang8 = ang6 - ang7;

cout << "Difference of two angles: ";

ang8.result();

}

else if (op == 5) { // Отношение углов

Angle ang9, ang10, ang11;

ang9.getangle();

ang10.getangle();

ang11 = ang9 / ang10;

cout << "Result of dividing one angle by another: " << ang11.returning() << endl;

}

else if (op == 6) { // Сравнение углов

Angle ang13, ang14;

ang13.getangle();

ang14.getangle();

if (ang13 == ang14)

cout << (ang13.returning() / 60) << " == " << (ang14.returning() / 60) << endl;

if (ang13 >= ang14)

cout << (ang13.returning() / 60) << " >= " << (ang14.returning() / 60) << endl;

if (ang13 <= ang14)

cout << (ang13.returning() / 60) << " <= " << (ang14.returning() / 60) << endl;

if (ang13 > ang14)

cout << (ang13.returning() / 60) << " > " << (ang14.returning() / 60) << endl;

if (ang13 < ang14)

cout << (ang13.returning() / 60) << " < " << (ang14.returning() / 60) << endl;

if (ang13 != ang14)

cout << (ang13.returning() / 60) << " != " << (ang14.returning() / 60) << endl;

}

else if (op == 7) { // Вычисление тригонометрических функций

Angle ang12;

ang12.getangle();

trig();

cout << "Select the trigonometric function you want to calculate:\n";

int m;

cin >> m;

while (m >= 1 && m <= 4) {

if (m == 1) {

cout << "Sinus: " << sin(ang12.radian()) << endl;

m = 0;

}

else if (m == 2) {

cout << "Cosinus: " << cos(ang12.radian()) << endl;

m = 0;

}

else if (m == 3) {

cout << "Tangents: " << tan(ang12.radian()) << endl;

m = 0;

}

else if (m == 4) {

cout << "Cotangents: " << 1 / tan(ang12.radian()) << endl;

m = 0;

}

else {

cout << "Error!"; // Ошибка при неверном выборе пункта меню

m = 0;

}

}

}

else if (op == 8) // Выход из программы

break;

else

cout << "Error!"; // Ошибка при неверном выборе пункта меню

cout << "Select menu item:\n";

cin >> op;

}

return 0;

}

**Вывод**

Изучила системы сборки на языке C++, изучила системы контроля версии, изучила основы работы с классами в С++ на основе класса Angle для работы с углами на плоскости, задаваемыми величиной в градусах и минутах.

**Список литературы**

1. BestProg - Programming: theory and practice/Электронный диплом/Режим доступа: <https://www.bestprog.net/ru/2018/04/03/c-classes-part-2-copy-constructor-examples-of-using-passing-a-class-object-to-a-function-returning-a-class-from-a-function_ru/#q01> (дата обращения: 16.09.2020).
2. Р. Лафоре, Объектно - ориентированное программирование в С++, 4-е издание. -М.: Питер, 2004 - 924 с. (дата обращения: 16.09.2020).