

Московский Авиационный Институт  
(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики  
Кафедра вычислительной математики и программирования

**Отчёт по Лабораторной работе №1**  
**“Простые Классы“**  
**по курсу “Объектно-Объективное Программирование“**  
**III Семестр**

Студент:	Катермин В.С.
Группа:	М8О-208Б-18
Преподаватель:	Журавлёв А.А.
Оценка:	
Дата:	<b>16.09.19</b>

1. **Тема:** Простые классы в C++.

2. **Код программы:**

**angle.h**

```
#ifndef D_ANGLE_H
#define D_ANGLE_H

#include <iostream>

struct Angle{
private:
    int deg;
    int min;

public:
    void read(std::istream& is);
    void awrite(std::ostream& os) const;
    int mntsonly();

    double deg_to_rad();

    int to360();

    Angle add(const Angle& rhs) const; // const в конце - нельзя изменять переменные именно этой структуры, на
    остальные (к примеру Angle result) не влияет, если они сами не константы
    Angle sub(const Angle& rhs) const;
    double div(const Angle& rhs) const;

    double asin();
    double acos();
    double atan();

    int comp(const Angle& rhs) const;
};

#endif
```

**angle.cpp**

```
#include "angle.h"
#include <cmath>

#include <iostream>

void Angle::read(std::istream& is) {
    is >> deg >> min;
}

void Angle::awrite(std::ostream& os) const {
    os << deg << ' ' << min << '\n';
}

int Angle::mntsonly(){
    if (deg < 0){
        return 60 - abs(min);
    } else {
        return min;
    }
}

double Angle::deg_to_rad(){
    double rad = ((double)deg + ((double)min / 60)) / 180 * M_PI; //M_PI - Пи в cmath
    //(double)angle - перевод angle (заданный в int) из int в double
    return rad;
}
```

```
}
```

```
int Angle::to360(){  
    if (deg < 0) {  
        return 360 + (deg % 360);  
    } else {  
        return deg % 360;  
    }  
}
```

```
Angle Angle::add(const Angle& rhs) const {  
    Angle result;  
    result.deg = deg + rhs.deg;  
    result.min = min + rhs.min;  
    if (result.min >= 60) {  
        result.deg += 1;  
        result.min -= 60;  
    } else if (result.min <= -60) {  
        result.deg -= 1;  
        result.min += 60;  
    }  
    return result;  
}
```

```
Angle Angle::sub(const Angle& rhs) const {  
    Angle result;  
    result.deg = deg - rhs.deg;  
    result.min = min - rhs.min;  
    if (result.min < 0) {  
        result.deg -= 1;  
        result.min += 60;  
    } else if ((result.deg < 0) && (result.min > 0)) {  
        result.deg += 1;  
        result.min -= 60;  
    }  
    return result;  
}
```

```
double Angle::div (const Angle& rhs) const {  
    double result = ((double)deg + ((double)min / 60)) / ((double)rhs.deg + ((double)rhs.min / 60));  
    return result;  
}
```

```
double Angle::asin() {  
    return sin((double)deg + ((double)min / 60));  
}
```

```
double Angle::acos() {  
    return cos((double)deg + ((double)min / 60));  
}
```

```
double Angle::atan(){  
    return tan((double)deg + ((double)min / 60));  
}
```

```
int Angle::comp(const Angle& rhs) const {  
    int result = 0;  
    if (deg > rhs.deg){  
        result = 1;  
    } else if (deg < rhs.deg) {  
        result = -1;  
    } else if (deg == rhs.deg) {  
        if (min > rhs.min) {
```

```

        result = 1;
    } else if (min < rhs.min) {
        result = -1;
    }
}
return result;
}

```

## lab1.cpp

```

#include <iostream>

#include "angle.h"

int main() {
    Angle a;
    Angle b;
    //std::cout << "Enter numbers:\n[degrees] and [minutes] (-59...59)" << std::endl;
    a.read(std::cin);
    b.read(std::cin);
    std::cout << "RADS:\n";
    std::cout << a.deg_to_rad() << std::endl;
    std::cout << "0-360 DEG:\n";
    std::cout << a.to360() << " " << a.mntsonly() << std::endl;
    std::cout << "SUM:\n";
    a.add(b).awrite(std::cout);
    std::cout << "SUB:\n";
    a.sub(b).awrite(std::cout);
    std::cout << "DIVIDE:\n";
    std::cout << a.div(b) << std::endl;
    std::cout << "SIN:\n";
    std::cout << a.asin() << std::endl;
    std::cout << "COS:\n";
    std::cout << a.acos() << std::endl;
    std::cout << "TAN:\n";
    std::cout << a.atan() << std::endl;

    switch (a.comp(b)) {
    case 0:
        std::cout << "A = B" << std::endl;
        break;

    case 1:
        std::cout << "A > B" << std::endl;
        break;

    case -1:
        std::cout << "A < B" << std::endl;
        break;
    }

    return 0;
}

```

### 3. Ссылка на репозиторий:

[https://github.com/GitGood2000/oop\\_projects](https://github.com/GitGood2000/oop_projects)

### 4. Набор testcases:

test\_00.test

30 16

20 47

**test\_00.result**

RADS:  
0.528253  
0-360 DEG:  
30 16  
SUM:  
51 3  
SUB:  
9 29  
DIVIDE:  
1.4563  
SIN:  
-0.912461  
COS:  
0.409163  
TAN:  
-2.23007  
 $A > B$

**test\_01.test**

361 23  
89 48

**test\_01.result**

RADS:  
6.30733  
0-360 DEG:  
1 23  
SUM:  
451 11  
SUB:  
271 35  
DIVIDE:  
4.02431  
SIN:  
-0.100011  
COS:  
-0.994986  
TAN:  
0.100515  
 $A > B$

**test\_02.test**

30 13  
60 43

**test\_02.result**

RADS:  
0.52738  
0-360 DEG:  
30 13  
SUM:  
90 56  
SUB:  
-30 30  
DIVIDE:  
0.497667  
SIN:  
-0.931771  
COS:  
0.363047  
TAN:  
-2.56653  
 $A < B$

**test\_03.test**

60 3

60 3

**test\_03.result**

RADS:

1.04807

0-360 DEG:

60 3

SUM:

120 6

SUB:

0 0

DIVIDE:

1

SIN:

-0.35203

COS:

-0.935989

TAN:

0.376106

A = B

**test\_04.test**

-293 -43

-142 -51

**test\_04.result**

RADS:

-5.12632

0-360 DEG:

67 17

SUM:

-436 -34

SUB:

-150 -52

DIVIDE:

2.05612

SIN:

0.999753

COS:

-0.0222446

TAN:

-44.9436

A &lt; B

**5. Результаты выполнения тестов:**

```
user@PSB133S01ZFH:~/3sem_projects/oop_projects/tests$ bash test.sh ../build/lab1
```

Test test\_00.test: SUCCESS

Test test\_01.test: SUCCESS

Test test\_02.test: SUCCESS

Test test\_03.test: SUCCESS

Test test\_04.test: SUCCESS

**6. Объяснение результатов работы программы:**

- 1) Программа вводит данные двух углов (a, b) конвертирует их из углов/минут в митуты
- 2) Программа выводит значение угла a в радианах
- 3) Программа выводит значение угла a в диапазоне от 0 до 360 градусов в градусах/минутах
- 4) Программа выводит результат сложения углов a и b в градусах/минутах
- 5) Программа выводит результат вычитания углов a и b в градусах/минутах
- 6) Программа выводит результат деления углов a и b
- 7) Программа выводит синус, косинус и тангенс угла a
- 8) Программа выводит результат проверки отношения равенства углов a и b.

7. **Вывод:** 1) Ознакомились с простыми классами в C++ и усвоили навык работы с ними; 2) Написана программа, производящая операции с простым классом.