

**Санкт-Петербургский государственный
университет телекоммуникаций
им. Профессора М.А. Бонч-Бруевича
Факультет ИСиТ**

Дисциплина: Технологии программирования

Отчёт по лабораторной работе №1
Проектирование классов и создание объектов

Выполнил:
студент группы ИСТ-666
Ерохин Б. А.

Проверил:
Медведев В. А.

Цель работы

Получение практических навыков проектирования классов, создания объектов, разработки и отладки объектно-ориентированных программ.

Задание

А) Спроектировать класс `Time`, содержащий две закрытые переменные (часы, минуты) и функции работы с ними. В состав функций (методов) должны входить:

- три конструктора: первый – без параметров, второй – с параметрами, третий – для создания копии объекта;
- деструктор;
- функции доступа к закрытым членам класса;
- функция, выводящая на экран показания времени в определённом формате;
- дружественные функции, увеличивающие значения часов и минут;

В) Создать три объекта, продемонстрировать работу функций.

Код программы

Содержимое файла **main.h**:

```
#include <stdio.h>
#include <time.h>

class Time {
public:
    Time();
    Time(int hours, int minutes);
    Time(const Time &obj);
    ~Time();

    void setHours(int value);
    void setMinutes(int value);

    friend void addHours(int value, Time &obj);
    friend void addMinutes(int value, Time &obj);

    int getHours();
    int getMinutes();

    void printTime();

private:
    int hours, minutes;
};
```

Содержимое файла **main.cpp**:

```
#include "main.h"

int main()
{
    Time system_time;
    system_time.setMinutes(11);
    system_time.printTime();

    Time other_time(20, 29);
    other_time.printTime();

    Time copy_time = other_time;
    addHours(6, copy_time);
    addMinutes(64, copy_time);
    copy_time.printTime();
    return 0;
}

Time::Time()
{
    time_t current_time;
    time(&current_time);

    struct tm *timeinfo = localtime(&current_time);
    setHours(timeinfo->tm_hour);
    setMinutes(timeinfo->tm_min);

    printf("Create %x\n", this);
}

Time::Time(int hours, int minutes)
{
    setHours(hours);
    setMinutes(minutes);

    printf("Create %x\n", this);
}

Time::Time(const Time &obj)
{
    setHours(obj.hours);
    setMinutes(obj.minutes);

    printf("Create %x\n", this);
}

Time::~~Time()
{
    printf("Delete %x\n", this);
}

void Time::setHours(int value)
{
    if (value < 0 || value > 24) {
        value = 0;
    }
    hours = value;
}
```

```

void Time::setMinutes(int value)
{
    if (value < 0 || value > 60) {
        value = 0;
    }
    minutes = value;
}

void addHours(int value, Time &obj)
{
    if (value + obj.getHours() < 0) {
        value = -obj.getHours();
    } else if (value + obj.getHours() > 24) {
        value -= 24;
    }
    obj.setHours(obj.getHours() + value);
}

void addMinutes(int value, Time &obj)
{
    if (value + obj.getMinutes() < 0) {
        addHours(-1, obj);
        value += 60;
        addMinutes(value, obj);
    } else if (value + obj.getMinutes() > 60) {
        addHours(1, obj);
        value -= 60;
        addMinutes(value, obj);
    } else {
        obj.setMinutes(obj.getMinutes() + value);
    }
}

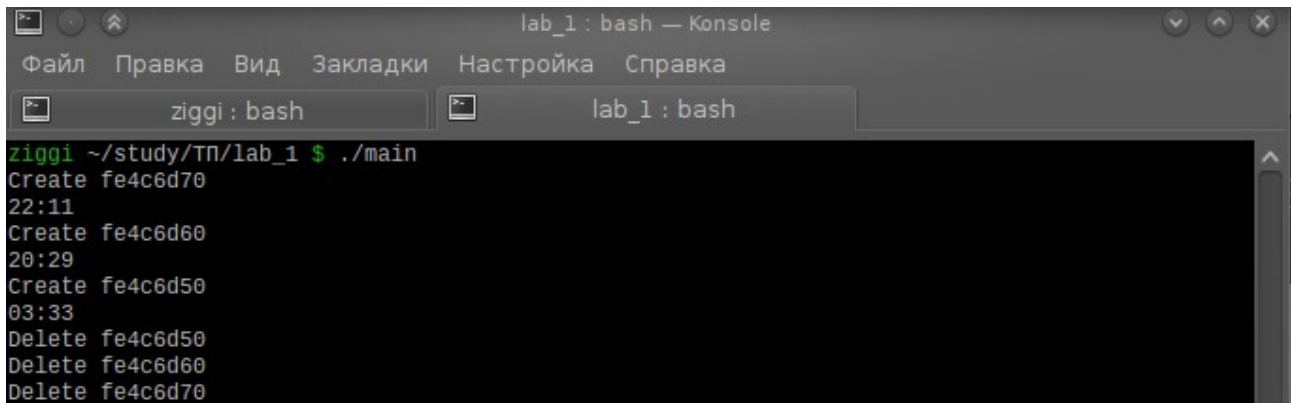
int Time::getHours()
{
    return hours;
}

int Time::getMinutes()
{
    return minutes;
}

void Time::printTime()
{
    printf("%02d:%02d\n", getHours(), getMinutes());
}

```

Результат работы программы



```
lab_1 : bash — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
ziggi : bash  lab_1 : bash
ziggi ~/study/ТП/lab_1 $ ./main
Create fe4c6d70
22:11
Create fe4c6d60
20:29
Create fe4c6d50
03:33
Delete fe4c6d50
Delete fe4c6d60
Delete fe4c6d70
```

Вывод

При запуске программы создаётся экземпляр класса Time под названием system_time, вызывается конструктор без параметров, в результате этого время устанавливается системное.

Далее создаётся экземпляр класса Time под названием other_time, вызывается конструктор с параметрами (час и минута), устанавливается время, которое было передано в конструктор.

После создаётся копия объекта other_time под названием copy_time, данные полностью копируются из этого объекта.