

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МОЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №1**  
**по дисциплине «Программирование»**  
**Тема: Работа с Make**

Студент гр. 6304

\_\_\_\_\_

Курков Д. В.

Преподаватель

\_\_\_\_\_

Берленко Т.А.

Санкт-Петербург

2016

## Цель работы

Освоить базовые принципы работы с утилитой make, создать Makefile для проекта.

## Задание

1. Создайте проект, состоящий из пяти файлов: **main.c**, **print\_str.c**, **get\_name.c**, **print\_str.h**, **get\_name.h** в каталоге, имя которого содержит Ваше имя, Фамилию и номер лабораторной.

- Файл **get\_name.c** должен содержать описание функции, которая считывает из входного потока имя пользователя и возвращает его.
- Файл **get\_name.h** должен содержать прототип функции, которая считывает из входного потока имя пользователя и возвращает его.
- Файл **print\_str.c** должен содержать описание функции, которая принимает в качестве аргумента строку и выводит её (функция ничего не возвращает).
- Файл **print\_str.h** должен содержать прототип функции, которая принимает в качестве аргумента строку и выводит её (функция ничего не возвращает).
- Файл **main.c** содержит главную функцию, которая вызывает функцию из файла **get\_name.h**, добавляет к результату выполнения функции строку “Hello,” и передает полученную строку в функцию вывода строки из **print\_str.h**.

2. После того, как Ваш проект будет готов, создайте для него Makefile.

## Выполнение задания

### 1. Создание проекта

1. Функция для считывающая из входного потока имя пользователя

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

char* get_name()
{
    char* name = (char*)malloc(80*sizeof(char));
    int i = 0;
    char ch;
    while ((ch = getchar()) != '\n')
    {
        name[i] = ch;
        i++;
    }
    name[i] = '\0';
    return name;
}
```

2. Функция для вывода строки, полученной на вход в качестве аргумента

```
void print_str (char str[])
{
    printf ("%s\n", str);
}
```

3. Функция main

```
int main()
{
    char hello[90] = "Hello, ";
    char* result;
    result = get_name();
    print_str(strncat(hello, result, 80));
    free(result);
    return 0;
}
```

4. Создание файлов get\_name.c, get\_name.h, print\_str.c, print\_str.h, main.c, содержащие исходный код, вышеописанных функций, их объявления и в заголовочные файлы.

## 2. Создание Makefile

Используя файлы из пункта 1.4, создаем makefile.

```
all: program clean

program: main.o get_name.o print_str.o
        gcc main.o get_name.o print_str.o -o program

get_name.o: get_name.c
        gcc -c get_name.c

print_str.o: print_str.c
        gcc -c print_str.c

main.o: main.c
        gcc -c main.c

clean:
        rm *.o
```

Где

**all: program clean** — конечная цель с условием ее достижения;

**clean: program** — удаление ненужных объектных файлов  
rm \*.o

**laba: main.o get\_name.o print\_str.o** — линковка, создание  
gcc main.o get\_name.o print\_str.o -o program исполняемого файла

**get\_name.o: get\_name.c** — компиляция,  
gcc -c get\_name.c создание объектного  
файла get\_name.o

описание строк, начинающихся с print\_str.o и main.o аналогично get\_name.c

**Вывод:** *В ходе данной лабораторной работы был создан проект, отвечающий условиям задания, создан Makefile для него, а также освоены особенности и возможности в работе с утилитой make.*