**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №1**

**по дисциплине «Программирование»**

Тема: Создание Makefile и управление кодом с помощью git.

|  |  |  |
| --- | --- | --- |
| Студентка гр. 6304 |  | Прозорова А.Д. |
| Преподаватель |  | Кринкин К.В. |

Санкт-Петербург

2016

**Цель**

Научиться создавать makefile и управлять кодом с помощью git.

**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.

**Ход работы**

*1. Makefile*

main: get\_name.o print\_str.o main.o

gcc main.o -o main get\_name.o print\_str.o

#запись результата в исполняемый файл name

rm \*.o

#очистка

main.o: main.c get\_name.h print\_str.h

gcc -c main.c

#cоздание объектного файла main.o

get\_name.o: get\_name.c get\_name.h

gcc -c get\_name.c

#cоздание объектного файла get\_name.o

print\_str.o: print\_str.c print\_str.h

gcc -c print\_str.c

#cоздание объектного файла print\_str.o

*2. Порядок работы с git*

* git clone https://github.com/moevm/pr1-2016-6304 — склонировать репозиторий группы 6304;
* git checkout –b Prozorova\_Anastasia\_L\_1 локально создать новую ветку от ветки master для нашего проекта;
* cp -r ~/»Рабочий стол»/lab\_1 ~/pr1-2016-6304/Prozorova\_Anastasia\_L\_1/lab\_1 — скопировать папку с лабораторной в личную ветку репозитория;
* git add lab\_1 — подтвердить изменения;

* git commit –m “#Prozorova\_Anastasia\_L\_1 lab1 added” - прокомментировать изменения;
* git push origin Prozorova\_Anastasia\_L\_1 - отправить изменения в репозиторий в личную ветку;
* создать pull request в ветку master.

**Вывод**

Цель работы достигнута, навыки работы с git и создания makefile освоены.

**Приложение 1**

**get\_name.c**

#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++;

}

return name;

}

**print\_str.c**

#include <stdio.h>

void print\_str(char\* name) {

printf ("%s", name);

}

**main.c**

#include <stdio.h>

#include <string.h>

#include "print\_str.h"

#include "get\_name.h"

#include <stdlib.h>

int main(){

char hello[90] = "Hello, ";

char\* result;

result = get\_name();

print\_str(strncat(hello, result, 80));

free(result);

return 0;

}