**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 проекта и работа на GitHub

|  |  |  |
| --- | --- | --- |
| Студент гр. 6304 |  | Ковынев М. В. |
| Преподаватель |  | Кринкин К. В. |

Санкт-Петербург

2016

## **Лабораторная работа №1**

## **Работа на сервере**

### **Цель работы:**

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.

### **Порядок проведения работы:**

**1.** Создаем проект, состоящий из пяти файлов: main.c, print\_str.c, get\_name.c , print\_str.h,т get\_name.h в каталоге, который содержит моё имя и фамилию и номер лабораторной. Файлы содержат следующие функции:

* Файл get\_name.c содержит описание функции, которая считывает из входного потока имя пользователя и возвращает его.
* Файл get\_name.h содержит прототип функции, которая считывает из входного потока имя пользователя и возвращает его.
* Файл print\_str.c содержит описание функции, которая принимает в качестве аргумента строку и выводит её (функция ничего не возвращает).
* Файл print\_str.h содержит 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 проект готов, создаем для него **MakeFile.**

**3.** Работа с репозиторием:

1. Скопируем репозиторий группы 6304

*Git clone https://github.com/moevm/pr1-2016-6304.git*

1. От ветки master создаем локальную ветку с названием работы и именем:

*Git checkout –b “Kovynev\_Maxim\_L\_1”*

1. В папку с клонированным репозиторием скопируем файлы с работой с помощью команды *cp –r*
2. Индексируем добавленные файлы командой

*git add \*.c \*.h makefile*

1. Коммитим наше изменений, чтобы подготовить их к отправке на сервер

*Git commit –m “Комментарий”*

1. Отправляем изменения в репозиторий командой

*Git push origin*

1. На сайте github создаем *pull request* в ветку *master.*

### **Вывод**

Освоил на практике создание makefile для сборки проекта, а так же работу с репозиторием Github.com и основные команды, связанные с индексированием изменений.

## **Приложение**

**Makefile**

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

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

rm \*.o

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

gcc -c main.c

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

gcc -c get\_name.c

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

gcc -c print\_str.c

### **Get\_name.c**

*#include <string.h>*

*#include <stdlib.h>*

*#include <stdio.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;*

*}*

### **Get\_name.h**

*char\* get\_name();*

### **Main.c**

*#include <stdio.h>*

*#include <stdlib.h>*

*#include <string.h>*

*#include "print\_str.h"*

*#include "get\_name.h"*

*int main(){*

*char hello[90] = "Hello, ";*

*char\* result;*

*result = get\_name();*

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

*free(result);*

*return 0;*

*}*

### **Print\_str.c**

#*include<stdio.h>*

*#include<string.h>*

*#include<stdlib.h>*

*char\* print\_str( char\* name)*

*{*

*printf("%s\n", name);*

*return 0;*

*};*

### **Print\_str.h**

*void print\_str( char\* str);*