

Отчет по лабораторной работе №13

Кашкин Иван Евгеньевич

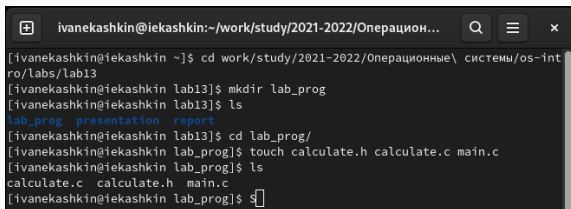
4 июня 2022 г.

РУДН, Москва, Россия

Приобрести простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования C калькулятора с простейшими функциями

Выполнить задание описанные в работе

Зашёл в каталог лабораторной работы, создал папку для программ и создал сами программы (рис. [-@fig:001])

A terminal window with a dark background. The title bar shows the user 'ivanekashkin@iekashkin' and the current directory '~/work/study/2021-2022/Операцион...'. The terminal contains the following commands and output:

```
[ivanekashkin@iekashkin ~]$ cd work/study/2021-2022/Операционные\ системы/os-intro/labs/lab13
[ivanekashkin@iekashkin lab13]$ mkdir lab_prog
[ivanekashkin@iekashkin lab13]$ ls
lab_prog  presentation  report
[ivanekashkin@iekashkin lab13]$ cd lab_prog/
[ivanekashkin@iekashkin lab_prog]$ touch calculate.h calculate.c main.c
[ivanekashkin@iekashkin lab_prog]$ ls
calculate.c calculate.h main.c
[ivanekashkin@iekashkin lab_prog]$
```

Рис. 1: Создание

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include "calculate.h"

float
Calculate(float Numeral, char Operation[4]){
    float SecondNumeral;
    if(strncmp(Operation,"+",1) == 0){
        printf("Второе слагаемое: ");
        scanf("%f", &SecondNumeral);
        return(Numeral+SecondNumeral);
    }
    else if (strncmp(Operation,"-",1) == 0){
        printf("Вычитаемое: ");
        scanf("%f",&SecondNumeral);
        return (Numeral+SecondNumeral);
    }
    else if (strncmp(Operation,"*",1) == 0){
        printf("Множитель: ");
        scanf("%f",&SecondNumeral);
        return(Numeral*SecondNumeral);
    }
    else if (strncmp(Operation,"/",1)==0){
        printf("Делитель: ");
        scanf("%f",&SecondNumeral);
        if(SecondNumeral == 0){
            printf("Ошибка деления на ноль!");
        }
        else
            return (Numeral/SecondNumeral);
    }
    else if(strncmp(Operation, "pow", 3) == 0)
    {
```

```
#ifndef CALCULATE_H_
#define CALCULATE_H_

float Calculate(float Numeral, char Operation[4]);

#endif
```

Рис. 4: calculate.h

```
#include <stdio.h>
#include "calculate.h"
int
main (void ){
    float Numeral;
    char Operation[4];
    float Result;
    printf ("Число: ");
    scanf("%f",&Numeral);
    printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
    scanf("%f",Operation);
    Result = Calculate(Numral,Operation);
    printf("%6.2f\n",Result);
    return 0;
}
```

Рис. 5: main.c

```
CC = gcc
CFLAGS = -g
LIBS = -lm

calcul: calculate.o main.o
    $(CC) calculate.o main.o -o calcul $(LIBS)

calculate.o: calculate.c calculate.h
    $(CC) -c calculate.c $(CFLAGS)

main.o: main.c calculate.h
    $(CC) -c main.c $(CFLAGS)

clean:
    -rm calcul *.o*
```

Рис. 6: Makefile

С помощью Makefile создал нужные файлы (рис. [-@fig:009])

```
[ivanekashkin@iekashkin lab_prog]$ make clean
rm calcul *.o*~
rm: невозможно удалить '*.o*~': Нет такого файла или каталога
make: [Makefile:15: clean] Ошибка 1 (игнорирование)
[ivanekashkin@iekashkin lab_prog]$ make calculate.o
make: «calculate.o» не требует обновления.
[ivanekashkin@iekashkin lab_prog]$ make calculate.o
gcc -c calculate.c
[ivanekashkin@iekashkin lab_prog]$ make main.o
gcc -c main.c
[ivanekashkin@iekashkin lab_prog]$ make calcul
make: *** Нет правила для сборки цели «calcul». Останов.
[ivanekashkin@iekashkin lab_prog]$ emacs &
[1] 11304
[ivanekashkin@iekashkin lab_prog]$ make calcul
gcc calculate.o main.o -o calcul -lm
[1]+  Завершён      emacs
[ivanekashkin@iekashkin lab_prog]$
```

Рис. 7: Make

```
[ivanekashkin@iekashkin lab_prog]$ gdb ./calcul
GNU gdb (GDB) Fedora 11.2-2.fc35
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-redhat-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./calcul...

This GDB supports auto-downloading debuginfo from the following URLs:
https://debuginfod.fedoraproject.org/
Enable debuginfod for this session? (y or [n]) y
Debuginfod has been enabled.
To make this setting permanent, add 'set debuginfod enabled on' to .gdbinit.
Downloading separate debug info for /home/ivanekashkin/work/study/2021-2022/Операционные системы/os-intro/labs/lab13/lab_prog/./calcul...
(No debugging symbols found in ./calcul)
(gdb) █
```

Рис. 8: Итог

В ходе выполнения данной лабораторной работы я приобрёл простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования C калькулятора с простейшими функциями

Спасибо за внимание!