

Средства, применяемые при разработке программного обеспечения в ОС типа UNIX/Linux

Федюшина Ярослава Андреевна

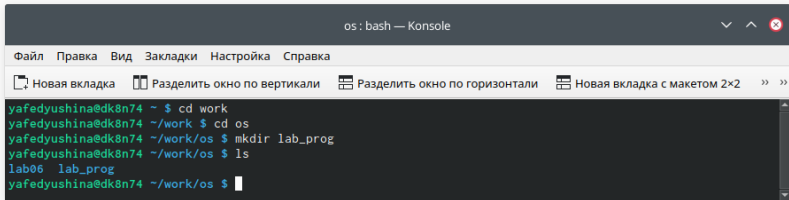
Российский Университет Дружбы Народов

Цель работы

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

Выполнение лабораторной работы

В домашнем каталоге создаю подкаталог `~/work/os/lab_prog`.



A terminal window titled "os : bash — Konsole" with a menu bar (Файл, Правка, Вид, Закладки, Настройка, Справка) and a toolbar (Новая вкладка, Разделить окно по вертикали, Разделить окно по горизонтали, Новая вкладка с макетом 2x2). The terminal shows the following commands and output:

```
yafedyushina@dk8n74 ~ $ cd work
yafedyushina@dk8n74 ~/work $ cd os
yafedyushina@dk8n74 ~/work/os $ mkdir lab_prog
yafedyushina@dk8n74 ~/work/os $ ls
lab06  lab_prog
yafedyushina@dk8n74 ~/work/os $
```

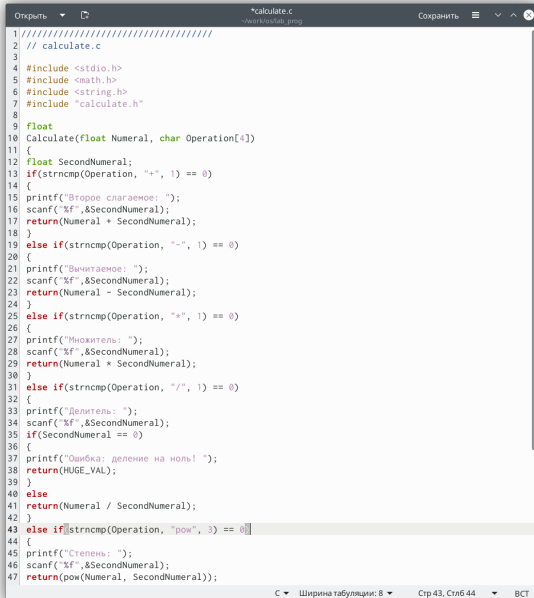
Figure 1: lab_prog

Создаю в нём файлы `calculate.h`, `calculate.c` и `main.c`

```
lab_prog : bash — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
[icon] Новая вкладка  [icon] Разделить окно по вертикали  [icon] Разделить окно по горизонтали  [icon] Новая вкладка с макетом 2x2  »  »
yafedyushina@dk8n74 ~/work/os $ cd lab_prog
yafedyushina@dk8n74 ~/work/os/lab_prog $ touch calculate.h calculate.c main.c
yafedyushina@dk8n74 ~/work/os/lab_prog $ ls
calculate.c calculate.h main.c
yafedyushina@dk8n74 ~/work/os/lab_prog $
```

Figure 2: создание файлов

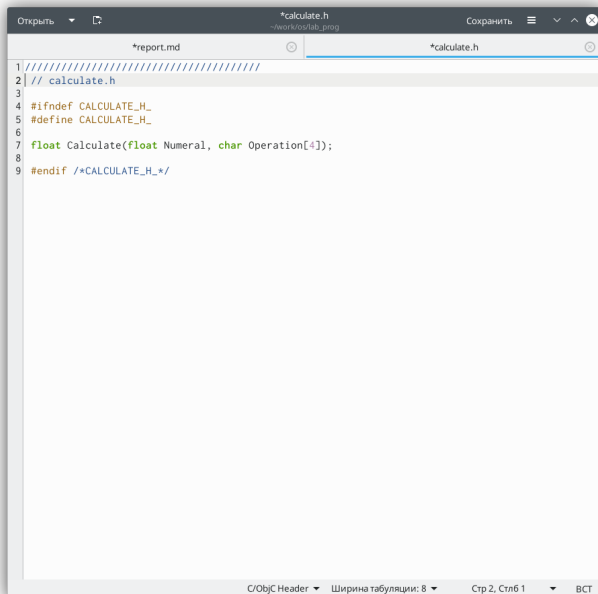
Реализация функций калькулятора в файле `calculate.h`



```
1 //////////////////////////////////////////////////
2 // calculate.c
3
4 #include <stdio.h>
5 #include <math.h>
6 #include <string.h>
7 #include "calculate.h"
8
9 float
10 Calculate(float Numeral, char Operation[4])
11 {
12     float SecondNumeral;
13     if(strncmp(Operation, "+", 1) == 0)
14     {
15         printf("Второе слагаемое: ");
16         scanf("%f", &SecondNumeral);
17         return(Numeral + SecondNumeral);
18     }
19     else if(strncmp(Operation, "-", 1) == 0)
20     {
21         printf("Вычитаемое: ");
22         scanf("%f", &SecondNumeral);
23         return(Numeral - SecondNumeral);
24     }
25     else if(strncmp(Operation, "*", 1) == 0)
26     {
27         printf("Множитель: ");
28         scanf("%f", &SecondNumeral);
29         return(Numeral * SecondNumeral);
30     }
31     else if(strncmp(Operation, "/", 1) == 0)
32     {
33         printf("Делитель: ");
34         scanf("%f", &SecondNumeral);
35         if(SecondNumeral == 0)
36         {
37             printf("Ошибка: деление на ноль! ");
38             return(HUGE_VAL);
39         }
40         else
41             return(Numeral / SecondNumeral);
42     }
43     else if(strncmp(Operation, "pow", 3) == 0)
44     {
45         printf("Степень: ");
46         scanf("%f", &SecondNumeral);
47         return(pow(Numeral, SecondNumeral));
```

Figure 3: calculate.h

Интерфейсный файл `calculate.h`, описывающий формат вызова функции калькулятора



The image shows a code editor window with two tabs: `*report.md` and `*calculate.h`. The `*calculate.h` tab is active, displaying the following C header file code:

```
1 //////////////////////////////////////////////////
2 // calculate.h
3
4 #ifndef CALCULATE_H_
5 #define CALCULATE_H_
6
7 float Calculate(float Numeral, char Operation[4]);
8
9 #endif /*CALCULATE_H_*/
```

The editor's status bar at the bottom indicates the file is a C/Object-C header, the tab width is 8, and the cursor is at line 2, column 1.

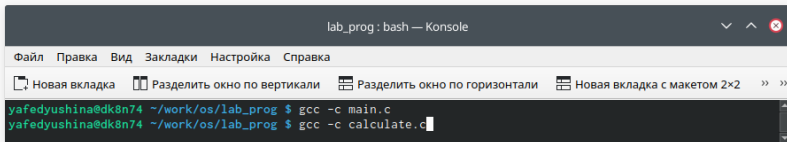
Figure 4: calculate.h

Основной файл `main.c`, реализующий интерфейс пользователя калькулятору

```
1 //////////////////////////////////////////////////
2 // main.c
3
4 #include <stdio.h>
5 #include "calculate.h"
6
7 int
8 main (void)
9 {
10 float Numeral;
11 char Operation[4];
12 float Result;
13 printf("Число: ");
14 scanf("%f",&Numeral);
15 printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): ");
16 scanf("%s",&Operation);
17 Result = Calculate(Numeral, Operation);
18 printf("%.2f\n",Result);
19 return 0;
20 }
```

Figure 5: main.c

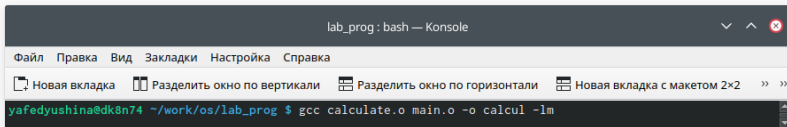
Далее выполняю компиляцию программы посредством gcc



A terminal window titled "lab_prog : bash — Konsole". The menu bar includes "Файл", "Правка", "Вид", "Закладки", "Настройка", and "Справка". The toolbar shows icons for "Новая вкладка", "Разделить окно по вертикали", "Разделить окно по горизонтали", and "Новая вкладка с макетом 2x2", followed by right arrow icons. The terminal text shows the user "yafedyushina@dk8n74" in the directory "~/work/os/lab_prog" executing two commands: `gcc -c main.c` and `gcc -c calculate.c`.

```
lab_prog : bash — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
[icon] Новая вкладка [icon] Разделить окно по вертикали [icon] Разделить окно по горизонтали [icon] Новая вкладка с макетом 2x2 » »
yafedyushina@dk8n74 ~/work/os/lab_prog $ gcc -c main.c
yafedyushina@dk8n74 ~/work/os/lab_prog $ gcc -c calculate.c
```

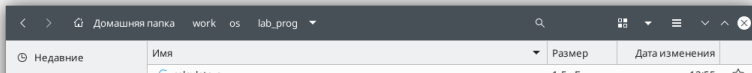
Figure 6: компиляция программы



A terminal window titled "lab_prog : bash — Konsole" with the same menu and toolbar as Figure 6. The terminal text shows the user "yafedyushina@dk8n74" in the directory "~/work/os/lab_prog" executing the command `gcc calculate.o main.o -o calcul -lm`.

```
lab_prog : bash — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
[icon] Новая вкладка [icon] Разделить окно по вертикали [icon] Разделить окно по горизонтали [icon] Новая вкладка с макетом 2x2 » »
yafedyushina@dk8n74 ~/work/os/lab_prog $ gcc calculate.o main.o -o calcul -lm
```

Figure 7: .



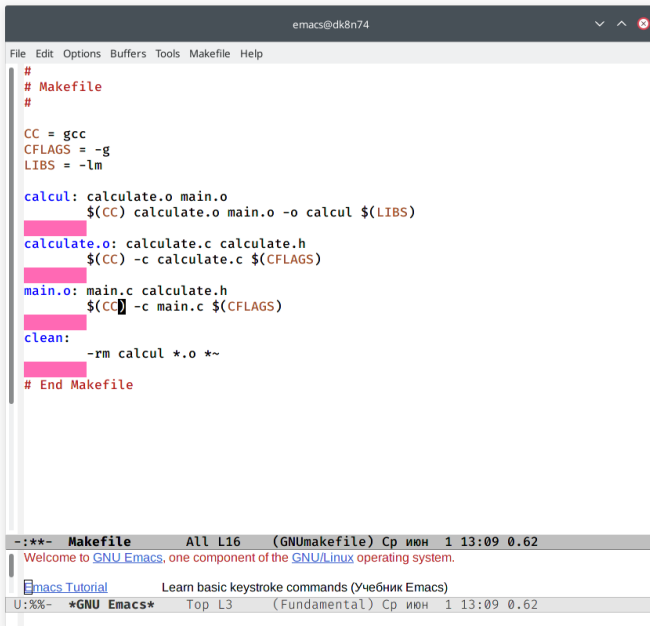
Создаю Makefile

```
1 #
2 # Makefile
3 #
4
5 CC = gcc
6 CFLAGS =
7 LIBS = -lm
8
9 calcul: calculate.o main.o
10     gcc calculate.o main.o -o calcul $(LIBS)
11
12 calculate.o: calculate.c calculate.h
13     gcc -c calculate.c $(CFLAGS)
14
15 main.o: main.c calculate.h
16     gcc -c main.c $(CFLAGS)
17
18 clean:
19     -rm calcul *.o *~
20
21 # End Makefile
```

Текст Ширина табуляции: 8 Стр 21, Стлб 15 ВСТ

Figure 9: код makefile

Далее исправляю код для Makefile



The image shows the Emacs editor interface. The title bar at the top reads "emacs@dk8n74". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Makefile", and "Help". The main text area contains a Makefile with the following content:

```
#
# Makefile
#

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 *~

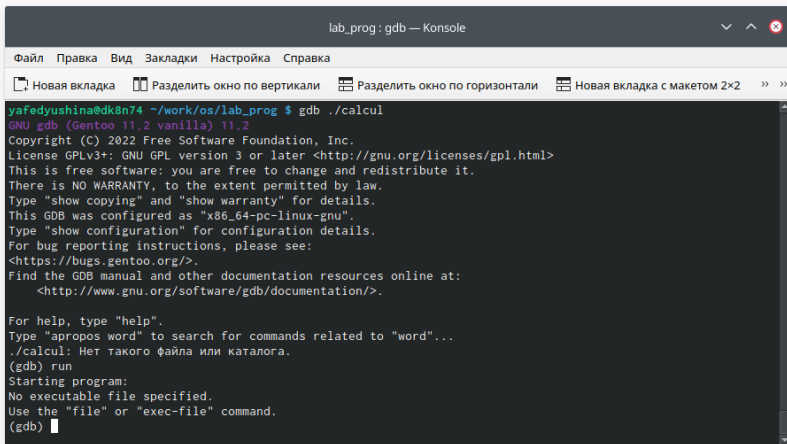
# End Makefile
```

At the bottom, the status bar displays the following information:

- Line 1, Column 16: `-:*** Makefile All L16 (GNUmakefile) Ср июн 1 13:09 0.62`
- Message: `Welcome to GNU Emacs, one component of the GNU/Linux operating system.`
- Help: `Emacs Tutorial Learn basic keystroke commands (Учебник Emacs)`
- Line 1, Column 3: `U:%%- *GNU Emacs* Top L3 (Fundamental) Ср июн 1 13:09 0.62`

Figure 10: исправление кода

С помощью gdb выполняю отладку программы calcul и запускаю отладчик GDB, загрузив в него программу для отладки. Для запуска ввожу команду run



```
lab_prog : gdb — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
[+] Новая вкладка  [||] Разделить окно по вертикали  [≡] Разделить окно по горизонтали  [≡] Новая вкладка с макетом 2x2  » »

yafedyushina@dk8n74 ~/work/os/lab_prog $ gdb ./calcul
GNU gdb (Gentoo 11.2 vanilla) 11.2
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-pc-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://bugs.gentoo.org/>.
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"...
./calcul: Нет такого файла или каталога.
(gdb) run
Starting program:
No executable file specified.
Use the "file" or "exec-file" command.
(gdb) █
```

Figure 11: запуск отладчика

Выводы

В ходе выполнения лабораторной работы №13 я приобрела простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования C калькулятора с простейшими функциями