

Лабораторная работа №13. Средства, применяемые при разработке программного обеспечения в ОС типа UNIX/Linux

Дисциплина: Операционные системы

Ганина Т. С.

2 мая 2023

Группа НКАбд-01-22

Российский университет дружбы народов, Москва, Россия

Информация

- Ганина Таисия Сергеевна
- Студентка 1-го курса, группа НКАбд-01-22
- Компьютерные и информационные науки
- Российский университет дружбы народов
- Ссылка на репозиторий гитхаба `tsganina`

Вводная часть

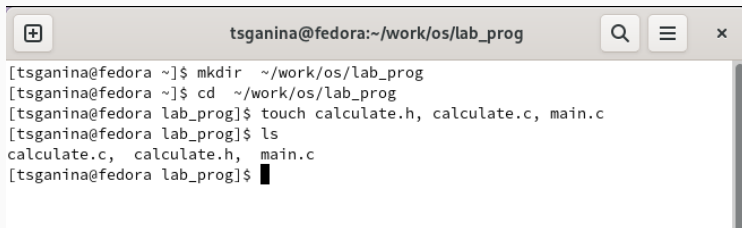
- Работа с программами позволит создать готовые скрипты для различных задач.

- Командный процессор ОС UNIX

- 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.

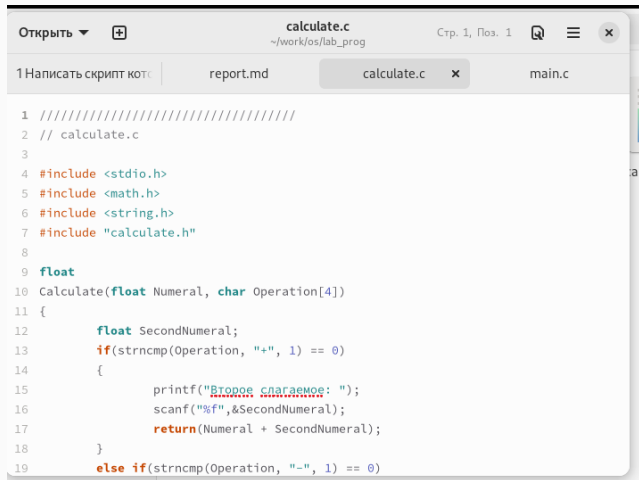
Выполнение заданий.

В домашнем каталоге создайте подкаталог `~/work/os/lab_prog`. Создайте в нём файлы: `calculate.h`, `calculate.c`, `main.c`

A terminal window with a title bar containing a plus icon, the text 'tsganina@fedora:~/work/os/lab_prog', a search icon, a menu icon, and a close icon. The terminal text shows a sequence of commands: 'mkdir ~/work/os/lab_prog', 'cd ~/work/os/lab_prog', 'touch calculate.h, calculate.c, main.c', and 'ls', which lists the created files. The prompt is '[tsganina@fedora lab_prog]\$' with a cursor.

```
[tsganina@fedora ~]$ mkdir ~/work/os/lab_prog
[tsganina@fedora ~]$ cd ~/work/os/lab_prog
[tsganina@fedora lab_prog]$ touch calculate.h, calculate.c, main.c
[tsganina@fedora lab_prog]$ ls
calculate.c, calculate.h, main.c
[tsganina@fedora lab_prog]$
```

Рис. 1: Создание файлов



```
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)
```

Рис. 2: calculate.c

Рис. 3: main.c

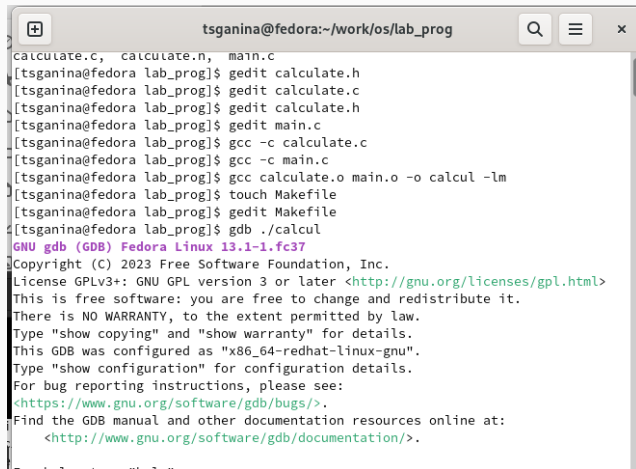


The image shows a code editor window with a tab titled 'calculate.h' at the top. The editor contains the following C header file code:

```
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_*/
```

Рис. 4: calculate.h

Выполните компиляцию программы посредством gcc. При необходимости исправьте синтаксические ошибки



```
tsganina@fedora:~/work/os/lab_prog
calculate.c, calculate.n, main.c
[tsganina@fedora lab_prog]$ gedit calculate.h
[tsganina@fedora lab_prog]$ gedit calculate.c
[tsganina@fedora lab_prog]$ gedit calculate.h
[tsganina@fedora lab_prog]$ gedit main.c
[tsganina@fedora lab_prog]$ gcc -c calculate.c
[tsganina@fedora lab_prog]$ gcc -c main.c
[tsganina@fedora lab_prog]$ gcc calculate.o main.o -o calcul -lm
[tsganina@fedora lab_prog]$ touch Makefile
[tsganina@fedora lab_prog]$ gedit Makefile
[tsganina@fedora lab_prog]$ gdb ./calcul
GNU gdb (GDB) Fedora Linux 13.1-1.fc37
Copyright (C) 2023 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/>.
```

Рис. 5: Компиляция

```
[tsganina@fedora lab_prog]$ touch Makefile
[tsganina@fedora lab_prog]$ gedit Makefile
[tsganina@fedora lab_prog]$ gdb ./calcul
GNU gdb (GDB) Fedora Linux 13.1-1.fc37
Copyright (C) 2023 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.
```

Рис. 6: Компиляция

С помощью gdb выполните отладку программы calcul (перед использованием gdb исправьте Makefile)

```
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 12
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 1
13.00
```

Рис. 7: Работа программы


```
[inferior 1 (process 6615) exited normally]
(gdb) list
1      //////////////////////////////////////////
2      // main.c
3
4      #include <stdio.h>
5      #include "calculate.h"
6
7      int
8      main (void)
9      {
10         float Numeral;
(gdb)
```

Рис. 8: list - несколько первых строк

```
(gdb) break 21
Breakpoint 1 at 0x40120f: file calculate.c, line 21.
(gdb) info breakpoints
Num      Type             Disp Enb Address                  What
1        breakpoint     keep y   0x000000000040120f in Calculate
                                                at calculate.c:21
(gdb) █
```

Рис. 9: Команда info breakpoints

```
[#####]5
Download failed: Таймер истек. Continuing without separate debug info for system-supplied DSO at 0x7ffff7fc6000.
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -

Breakpoint 1, Calculate (Numeral=5, Operation=0x7fffffffda54 "-")
    at calculate.c:21
21                                     printf("Вычитаемое: ");
(gdb)
```

Рис. 10: Работа программы с прерыванием

```
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: Операция (+,-,*,/,pow,sqrt,sin,cos,tan): -

Breakpoint 1, Calculate (Numeral=5, Operation=0x7fffffffda54 "-")
  at calculate.c:21
21                                     printf("Вычитаемое: ");
(gdb) backtrace
#0 Calculate (Numeral=5, Operation=0x7fffffffda54 "-") at calculate.c:21
#1 0x00000000004014eb in main () at main.c:17
(gdb) █
```

Рис. 11: backtrace

```
(gdb) print Numeral  
$1 = 5  
(gdb) █
```

Рис. 12: Значение переменной Numeral

```
#1  0x0000000000004014eb  III  III  III  I
(gdb) print Numeral
$1 = 5
(gdb) display Numeral
1: Numeral = 5
(gdb) █
```

Рис. 13: Та же переменная, но как бы после выполнения программы

```

(gdb) info breakpoints
Num      Type          Disp Enb Address                What
1        breakpoint    keep y  0x0000000000040120f in Calculate
                                     at calculate.c:21
        breakpoint already hit 1 time
2        breakpoint    keep y  0x0000000000040120f in Calculate
                                     at calculate.c:21
        breakpoint already hit 1 time
(gdb) delete 1
(gdb) info breakpoints
Num      Type          Disp Enb Address                What
2        breakpoint    keep y  0x0000000000040120f in Calculate
                                     at calculate.c:21
        breakpoint already hit 1 time
(gdb) delete 2
(gdb) info breakpoints
No breakpoints or watchpoints.
(gdb) █

```

Рис. 14: Удаление точек прерывания

Результаты

При выполнении данной лабораторной работы 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.