

# Презентация по лабораторной работе №13

Дисциплина: операционные системы

---

Лобанова П. И.

06.05.2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Лобанова Полина Иннокентьевна
- учащаяся факультета физико-математических и естественных наук
- учащаяся направления “Компьютерные и информационные науки”
- Российский университет дружбы народов
- [polla-2004@mail.ru](mailto:polla-2004@mail.ru)

## Цель работы

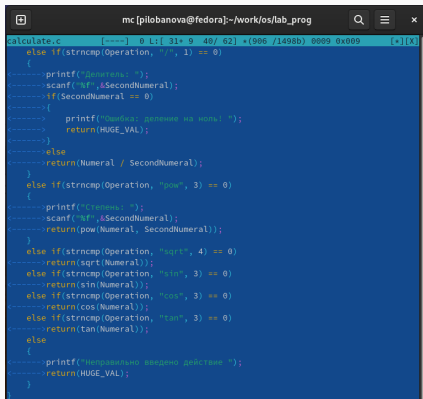
---

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.

## Ход работы

---

Первое, что нужно было сделать для создания калькулятора, это создание и заполнение всех необходимых файлов.



```
calculate.c 0 L: [ 31* 9 40/ 62] *(906 /1498b) 0009 0x009 [*](X)
else if(strncmp(Operation, "/", 1) == 0)
{
    printf("Ошибка: деление на ноль! ");
    return(HUGE_VAL);
}
else
{
    return(Numeral / SecondNumeral);
}
else if(strncmp(Operation, "pow", 3) == 0)
{
    printf("Ошибка: ");
    scanf("%f", &SecondNumeral);
    return(pow(Numeral, SecondNumeral));
}
else if(strncmp(Operation, "sqrt", 4) == 0)
{
    return(sqrt(Numeral));
}
else if(strncmp(Operation, "sin", 3) == 0)
{
    return(sin(Numeral));
}
else if(strncmp(Operation, "cos", 3) == 0)
{
    return(cos(Numeral));
}
else if(strncmp(Operation, "tan", 3) == 0)
{
    return(tan(Numeral));
}
else
{
    printf("Неправильно введено действие ");
    return(HUGE_VAL);
}
```

Рис. 1: Содержание одного из файлов.

Далее, нужно было запустить скомпилированный файл через отладчик, чтобы проверить его работу.

```
(gdb) run
Starting program: /home/pilobanova/work/os/lab_prog/calcul
Downloading 0.01 MB separate debug info for system-supplied DSO at 0x7ffff7fc4000
Downloading 2.25 MB separate debug info for /lib64/libm.so.6
Downloading 7.42 MB separate debug info for /lib64/libc.so.6
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib64/libthread_db.so.1".
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 6
11.00
[Inferior 1 (process 3051) exited normally]
```

Рис. 2: Результат работы программы.



С помощью отладчика мы смогли увидеть исходный код (или же только нужные нам строки), установить точки останова и многое другое.

```
(gdb) list
1      /* Terminate the frame unwind info section with a 4byte 0 as a sentinel;
2         this would be the 'length' field in a real FDE.  */
3
4      typedef unsigned int ui32 __attribute__((mode(SI)));
5      static const ui32 __FRAME_END__[1]
6      __attribute__((used, section(".eh_frame")))
7      = { 0 };
```

Рис. 3: Часть кода.

Последнее, что нам нужно было сделать, это проанализировать два файла с помощью команды splint.

```
[pi@banovafedora lab_prog]$ splint calculate.c
Splint 3.1.2 --- 22 Jan 2022

calculate.h:7:37: Function parameter Operation declared as manifest array (size
      constant is meaningless)
  A formal parameter is declared as an array with size. The size of the array
  is ignored in this context, since the array formal parameter is treated as a
  pointer. (Use -fixedformalarray to inhibit warning)
calculate.c:10:31: Function parameter Operation declared as manifest array
      (size constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:16:2: Return value (type int) ignored: scanf("%f", &sec...
  Result returned by function call is not used. If this is intended, can cast
  result to (void) to eliminate message. (Use -retvalint to inhibit warning)
calculate.c:22:2: Return value (type int) ignored: scanf("%f", &sec...
calculate.c:28:2: Return value (type int) ignored: scanf("%f", &sec...
calculate.c:34:2: Return value (type int) ignored: scanf("%f", &sec...
calculate.c:35:5: Dangerous equality comparison involving float types:
      SecondNumeral == 0
  Two real (float, double, or long double) values are compared directly using
  == or != primitive. This may produce unexpected results since floating point
  representations are inexact. Instead, compare the difference to FLT_EPSILON
  or DBL_EPSILON. (Use -realcompare to inhibit warning)
calculate.c:38:12: Return value type double does not match declared type float:
      (HUGE_VAL)
  To allow all numeric types to match, use -relaxtypes.
calculate.c:40:2: Return value (type int) ignored: scanf("%f", &sec...
calculate.c:47:8: Return value type double does not match declared type float:
      (pow(Numeral, SecondNumeral))
calculate.c:58:8: Return value type double does not match declared type float:
      (sort(Numeral))
calculate.c:52:8: Return value type double does not match declared type float:
      (sin(Numeral))
calculate.c:54:8: Return value type double does not match declared type float:
      (cos(Numeral))
calculate.c:56:8: Return value type double does not match declared type float:
      (tan(Numeral))
calculate.c:60:8: Return value type double does not match declared type float:
      (HUGE_VAL)
Finished checking --- 15 code warnings
```

Рис. 4: Анализ одного из файлов.

## Контрольные вопросы

---

1. Как получить информацию о возможностях программ gcc, make, gdb и др.?
2. Назовите и дайте краткую характеристику основным этапам разработки приложений в UNIX.
3. Что такое суффикс в контексте языка программирования? Приведите примеры использования.
4. Каково основное назначение компилятора языка C в UNIX?
5. Для чего предназначена утилита make?
6. Приведите пример структуры Makefile. Дайте характеристику основным элементам этого файла.

7. Назовите основное свойство, присущее всем программам отладки. Что необходимо сделать, чтобы его можно было использовать?
8. Назовите и дайте основную характеристику основным командам отладчика gdb.
9. Опишите по шагам схему отладки программы, которую Вы использовали при выполнении лабораторной работы.
10. Прокомментируйте реакцию компилятора на синтаксические ошибки в программе при его первом запуске.
11. Назовите основные средства, повышающие понимание исходного кода программы.
12. Каковы основные задачи, решаемые программой splint?

## Вывод

---

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.