

Лабораторная Работа № 14

Адебайо Ридвануллахи Айофе

Москва, 2021

Российский Университет Дружбы Народов

Операционные Системы

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.

1. Составьте каталог
2. Создание и заполнение файлов .c
3. Создайте файл Makefile
4. Используйте GDB для отладки и запуска файла .c
5. Используйте команду splint

- Составьте каталог

```
raadebayjo@dk6n54 ~ $ mkdir work
mkdir: невозможно создать каталог «work»: Файл существует
raadebayjo@dk6n54 ~ $ mkdir work/os
mkdir: невозможно создать каталог «work/os»: Файл существует
raadebayjo@dk6n54 ~ $ mkdir work/os/lab_prog
raadebayjo@dk6n54 ~ $ cd work/os/lab_prog
raadebayjo@dk6n54 ~/work/os/lab_prog $
```

Figure 1: каталог

- Создание и заполнение файлов .c

```
1 #include <stdio.h>
2 #include <math.h>
3 #include <string.h>
4 #include "calculate.h"
5 float
6 Calculate(float Numeral, char Operation[4])
7 {
8     float SecondNumeral;
9     if(strncmp(Operation, "+", 1) == 0)
10     {
```

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

Figure 3: файл

```
1 #include <stdio.h>
2 #include "calculate.h"
3
4 int
5 main (void)
6 {
7     float Numeral;
8     char Operation[4];
9     float Result;
10    printf("Число: ");
```

- Создайте файл 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 *~
```

- Используйте GDB для отладки и запуска файла .c

```
raadebayjo@dk6n54 ~/work/os/lab_prog $ gdb ./calcul
GNU gdb (Gentoo 10.1 vanilla) 10.1
Copyright (C) 2020 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"...
Reading symbols from ./calcul...
(No debugging symbols found in ./calcul)
(gdb) █
```

Figure 6: GDB

```
(gdb) list calculate.c:20,29
20         }
21         ...
```



```
(gdb) list calculate.c:20,27
20         }
21         else if(strncmp(Operation, "*", 1) == 0)
22         {
23             printf("Множитель: ");
24             scanf("%f",&SecondNumeral);
25             return(Numeral * SecondNumeral);
26         }
27         else if(strncmp(Operation, "/", 1) == 0)
(gdb) break21
Undefined command: "break21". Try "help".
(gdb) break 21
Breakpoint 1 at 0x991: file calculate.c, line 21.
(gdb)
```

Figure 8: GDB

- Используйте команду splint

```
Терминал - raadebayjo@dk6n65:~/work/os/lab_prog
Файл  Правка  Вид  Терминал  Вкладки  Справка
raadebayjo@dk6n65 ~/work/os/lab_prog $ splint calculate.c
Splint 3.1.2 --- 13 Jan 2021

calculate.h:4:37: 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:6:31: Function parameter Operation declared as manifest array (size
        constant is meaningless)
calculate.c: (in function Calculate)
calculate.c:12:3: 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:18:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:24:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:30:3: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:31:6: 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:34:10: Return value type double does not match declared type float:
```

В результате работы , я приобрёл простейшие навыки разработки, анализа, тестирования и отладки приложений в Линукс

Спасибо за внимание
