Средства, применяемые приразработке программного обеспечения в ОС типа UNIX/Linux

Джумаев Бегенч

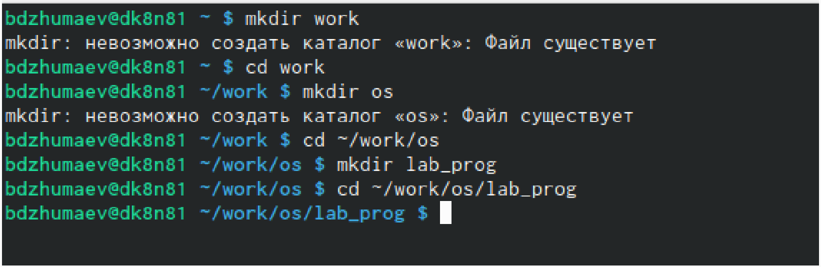
03.06.2021

Цель работы

Цель работы

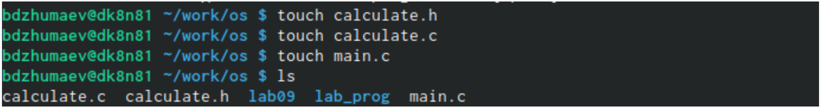
Приобрести простейшие навыки разработки, анализа, тестирования и отладкиприложений в ОС типа UNIX/Linux на примере создания на языке программирова-ния С калькулятора с простейшими функциями.

Создание подкаталога



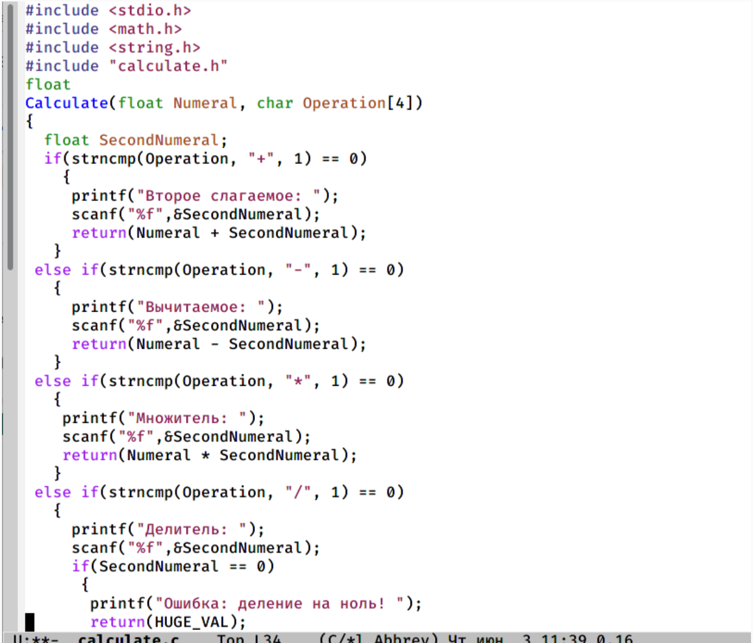
```
bdzhumaev@dk8n81 ~ $ mkdir work
mkdir: невозможно создать каталог «work»: Файл существует
bdzhumaev@dk8n81 ~ $ cd work
bdzhumaev@dk8n81 ~/work $ mkdir os
mkdir: невозможно создать каталог «os»: Файл существует
bdzhumaev@dk8n81 ~/work $ cd ~/work/os
bdzhumaev@dk8n81 ~/work/os $ mkdir lab_prog
bdzhumaev@dk8n81 ~/work/os $ cd ~/work/os/lab_prog
bdzhumaev@dk8n81 ~/work/os/lab_prog $
```

Создание файлов



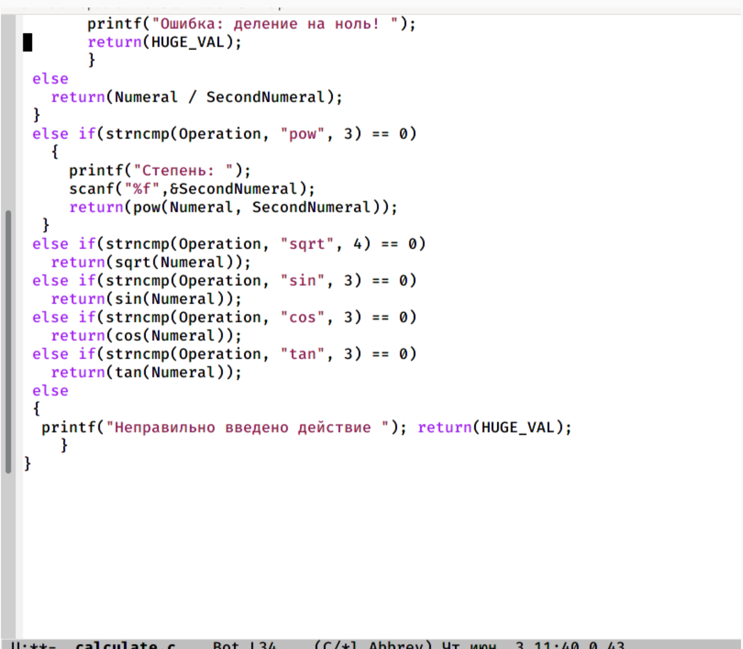
```
bdzhumaev@dk8n81 ~/work/os $ touch calculate.h
bdzhumaev@dk8n81 ~/work/os $ touch calculate.c
bdzhumaev@dk8n81 ~/work/os $ touch main.c
bdzhumaev@dk8n81 ~/work/os $ ls
calculate.c  calculate.h  lab09  lab_prog  main.c
```

Реализация функций калькулятора в файле calculate.c



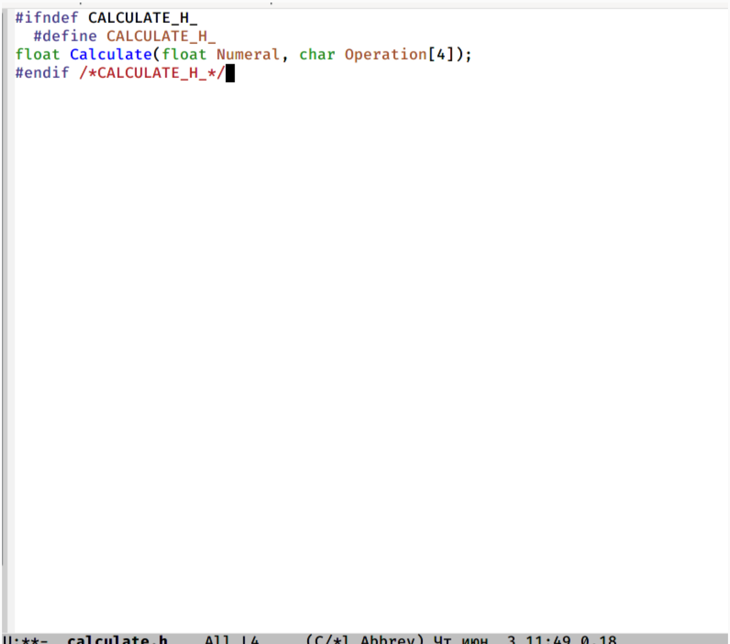
```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include "calculate.h"
float
Calculate(float Numeral, char Operation[4])
{
  float SecondNumeral;
  if(strncmp(Operation, "+", 1) == 0)
    {
     printf("Второе слагаемое: ");
     scanf("%f",&SecondNumeral);
     return(Numeral + SecondNumeral);
   }
 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: деление на ноль! ");
       return(HUGE_VAL);
U:**-  calculate.c    Top L34    (C/*l Abbrev) Чт июн  3 11:39 0.16
```

Реализация функций калькулятора в файле calculate.c



```
        printf("Ошибка: деление на ноль! ");
        return(HUGE_VAL);
        }
 else
   return(Numeral / SecondNumeral);
 }
 else if(strncmp(Operation, "pow", 3) == 0)
   {
     printf("Степень: ");
     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);
    }
}

U:**-  calculate.c    Bot L34    (C/*l Abbrev) Чт июн  3 11:40 0.43
```

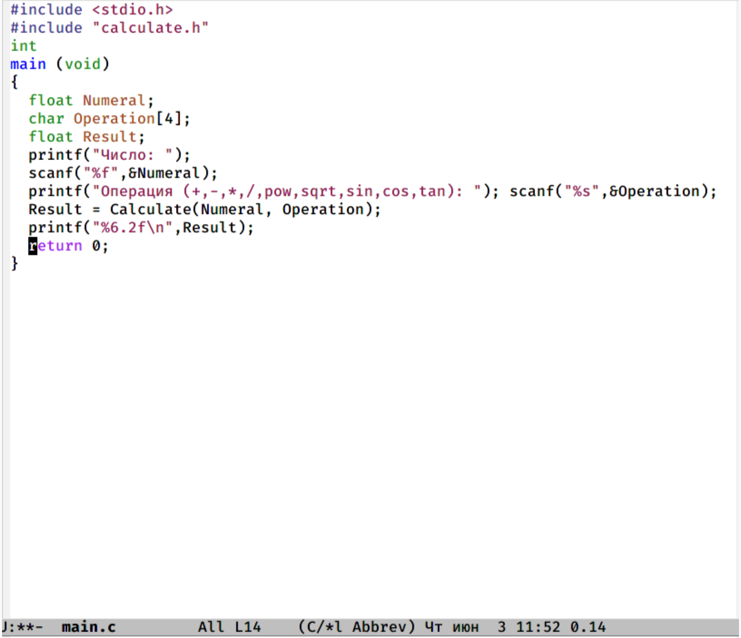
Интерфейсный файл calculate.h



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

```
U:**-  calculate.h    All L4     (C/*l Abbrev) Чт июн  3 11:49 0.18
```

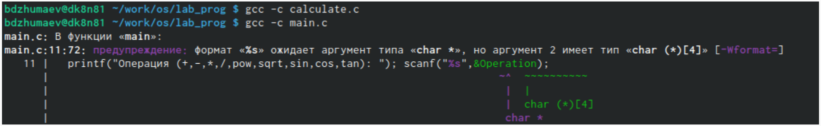
Основной файл main.c



```
#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("%s",&Operation);
  Result = Calculate(Numeral, Operation);
  printf("%6.2f\n",Result);
  return 0;
}
```

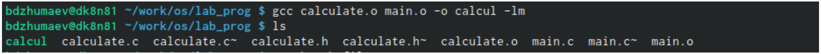
```
J:**-  main.c         All L14    (C/*l Abbrev) Чт июн  3 11:52 0.14
```

Команда gcc



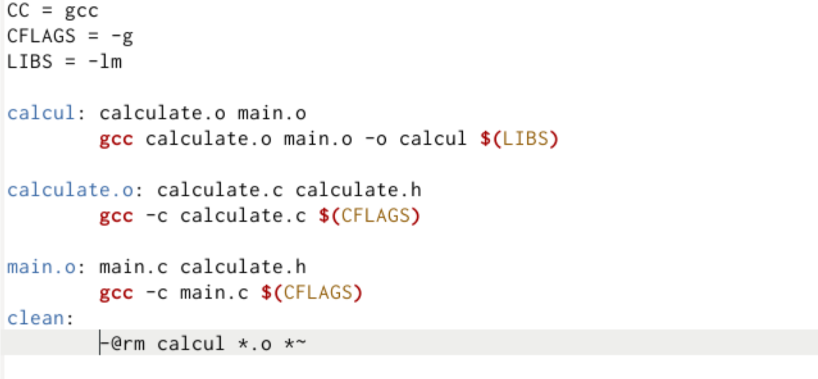
```
bdzhumaev@dk8n81 ~/work/os/lab_prog $ gcc -c calculate.c
bdzhumaev@dk8n81 ~/work/os/lab_prog $ gcc -c main.c
main.c: В функции «main»:
main.c:11:72: предупреждение: формат «%s» ожидает аргумент типа «char *», но аргумент 2 имеет тип «char (*)[4]» [-Wformat=]
   11 |   printf("Операция (+,-,*,/,pow,sqrt,sin,cos,tan): "); scanf("%s",&Operation);
      |                                                                       ~^   ~~~~~~~~~~
      |                                                                        |   |
      |                                                                        |   char (*)[4]
      |                                                                        char *
```

Компиляция



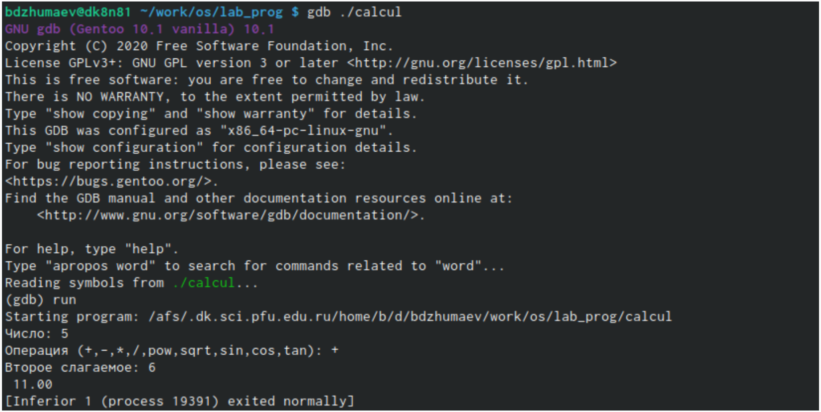
```
bdzhumaev@dk8n81 ~/work/os/lab_prog $ gcc calculate.o main.o -o calcul -lm
bdzhumaev@dk8n81 ~/work/os/lab_prog $ ls
calcul  calculate.c  calculate.c~  calculate.h  calculate.h~  calculate.o  main.c  main.c~  main.o
```

Makefile



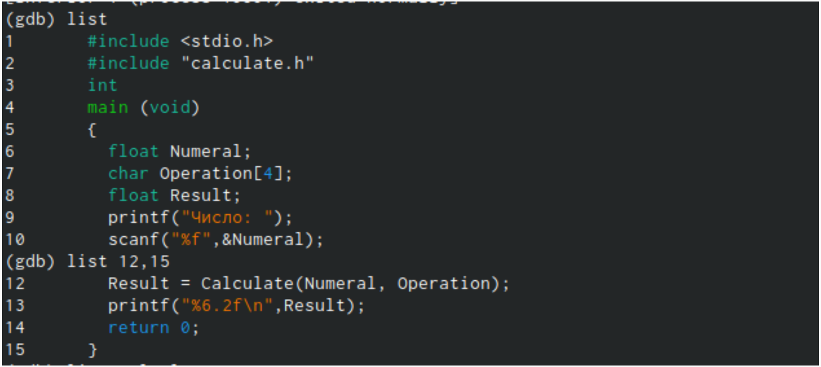
```
CC = gcc
CFLAGS = -g
LIBS = -lm

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

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

main.o: main.c calculate.h
	gcc -c main.c $(CFLAGS)
clean:
	-@rm calcul *.o *~
```

Отладчик GDB и run



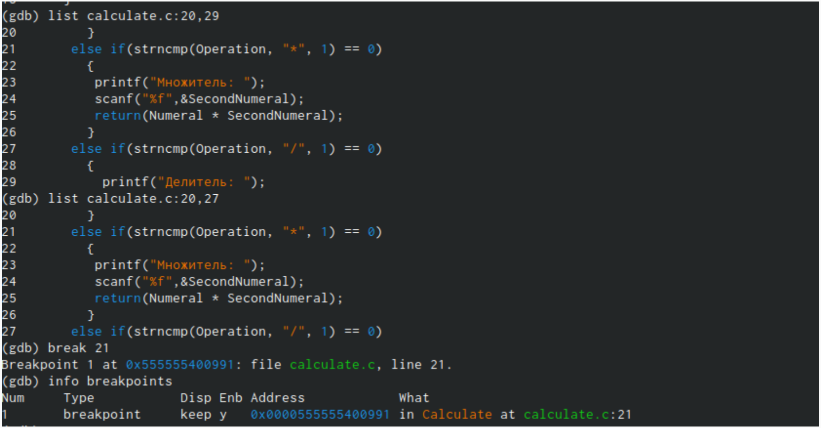
```
bdzhumaev@dk8n81 ~/work/os/lab_prog $ gdb ./calcul
GNU gdb (Gentoo 10.1 vanilla) 10.1
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, 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...
(gdb) run
Starting program: /afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/work/os/lab_prog/calcul
Число: 5
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 6
 11.00
[Inferior 1 (process 19391) exited normally]
```

Команда list



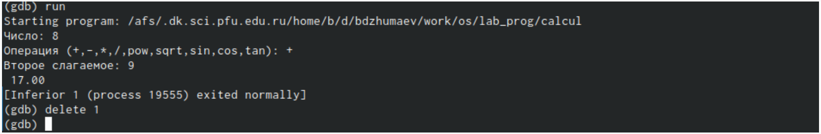
```
(gdb) list
1	#include <stdio.h>
2	#include "calculate.h"
3	int
4	main (void)
5	{
6	  float Numeral;
7	  char Operation[4];
8	  float Result;
9	  printf("Число: ");
10	  scanf("%f",&Numeral);
(gdb) list 12,15
12	  Result = Calculate(Numeral, Operation);
13	  printf("%6.2f\n",Result);
14	  return 0;
15	}
```

Команда list,точка остановка и информация



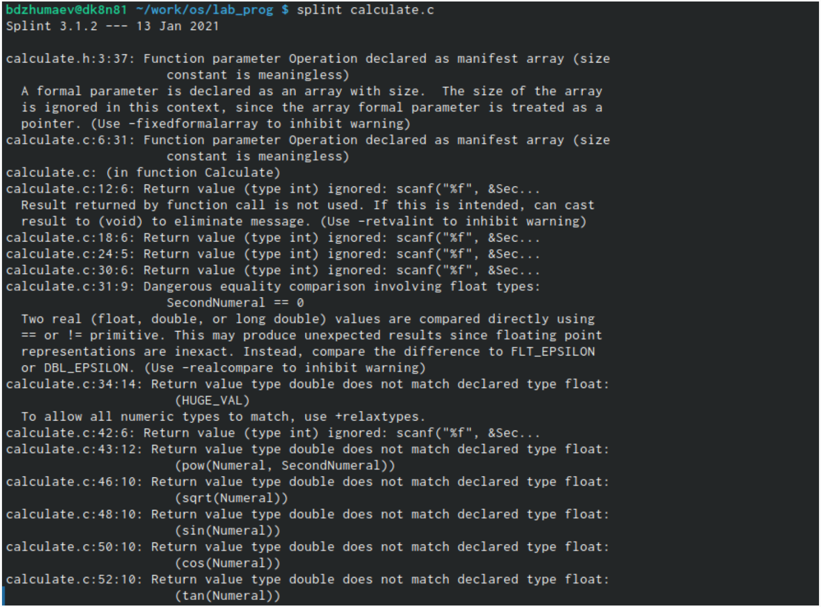
```
(gdb) list calculate.c:20,29
20          }
21        else if(strncmp(Operation, "*", 1) == 0)
22          {
23           printf("Множитель: ");
24           scanf("%f",&SecondNumeral);
25           return(Numeral * SecondNumeral);
26          }
27        else if(strncmp(Operation, "/", 1) == 0)
28          {
29            printf("Делитель: ");
(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) break 21
Breakpoint 1 at 0x555555400991: file calculate.c, line 21.
(gdb) info breakpoints
Num     Type           Disp Enb Address            What
1       breakpoint     keep y   0x0000555555400991 in Calculate at calculate.c:21
```

Run, команда backtrace, Numeral и удаление точки останова



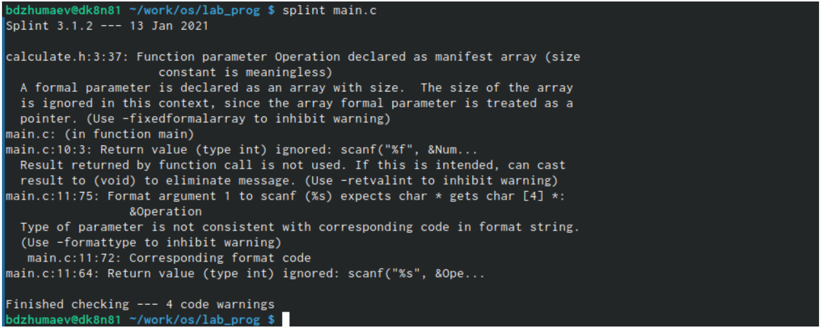
```
(gdb) run
Starting program: /afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/work/os/lab_prog/calcul
Число: 8
Операция (+,-,*,/,pow,sqrt,sin,cos,tan): +
Второе слагаемое: 9
 17.00
[Inferior 1 (process 19555) exited normally]
(gdb) delete 1
(gdb) 
```

Анализ calculate.c



```
bdzhumaev@dk8n81 ~/work/os/lab_prog $ splint calculate.c
Splint 3.1.2 --- 13 Jan 2021

calculate.h:3:37: 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:12:6: 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:18:6: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:24:5: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:30:6: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:31:9: 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:34:14: Return value type double does not match declared type float:
                      (HUGE_VAL)
  To allow all numeric types to match, use +relaxtypes.
calculate.c:42:6: Return value (type int) ignored: scanf("%f", &Sec...
calculate.c:43:12: Return value type double does not match declared type float:
                      (pow(Numeral, SecondNumeral))
calculate.c:46:10: Return value type double does not match declared type float:
                      (sqrt(Numeral))
calculate.c:48:10: Return value type double does not match declared type float:
                      (sin(Numeral))
calculate.c:50:10: Return value type double does not match declared type float:
                      (cos(Numeral))
calculate.c:52:10: Return value type double does not match declared type float:
                      (tan(Numeral))
```

Анализ main.c



```
bdzhumaev@dk8n81 ~/work/os/lab_prog $ splint main.c
Splint 3.1.2 --- 13 Jan 2021

calculate.h:3:37: 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)
main.c: (in function main)
main.c:10:3: Return value (type int) ignored: scanf("%f", &Num...
  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)
main.c:11:75: Format argument 1 to scanf (%s) expects char * gets char [4] *:
                 &Operation
  Type of parameter is not consistent with corresponding code in format string.
  (Use -formattype to inhibit warning)
   main.c:11:72: Corresponding format code
main.c:11:64: Return value (type int) ignored: scanf("%s", &Ope...

Finished checking --- 4 code warnings
bdzhumaev@dk8n81 ~/work/os/lab_prog $ 
```

Вывод

Изучал основы программирования в оболочке ОС UNIX, научилась писать бо- лее сложные командные файлы с использованием логических управляющих конструкций и циклов.