

# **Отчет по лабораторной работе №13**

**Дисциплина операционные системы**

Старикова Евгения Дмитриевна

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Выводы	13

## **Список иллюстраций**

## Список таблиц

# 1 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.

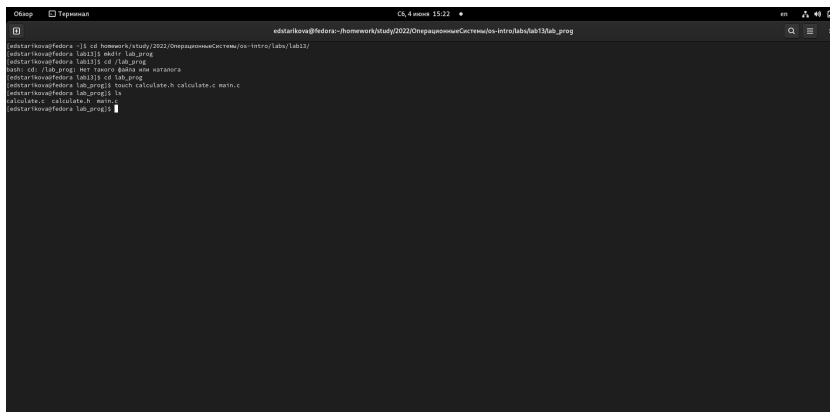
## 2 Задание

1. В домашнем каталоге создайте подкаталог `~/work/os/lab_prog`.
2. Создайте в нём файлы: `calculate.h`, `calculate.c`, `main.c`. Это будет примитивнейший калькулятор, способный складывать, вычитать, умножать и делить, возводить число в степень, брать квадратный корень, вычислять `sin`, `cos`, `tan`. При запуске он будет запрашивать первое число, операцию, второе число. После этого программа выведет результат и остановится.
3. Выполните компиляцию программы посредством `gcc`:
  - `gcc -c calculate.c`
  - `gcc -c main.c`
  - `gcc calculate.o main.o -o calcul -lm`
  - При необходимости исправьте синтаксические ошибки.
  - Создайте `Makefile`. Поясните в отчёте его содержание.
  - С помощью `gdb` выполните отладку программы `calcul` (перед использованием `gdb` исправьте `Makefile` ):
4. Запустите отладчик GDB, загрузив в него программу для отладки
5. Для запуска программы внутри отладчика введите команду `run`
6. Для постраничного (по 10 строк) просмотра исходного кода используйте команду `list`
  - Для просмотра строк с 12 по 15 основного файла используйте `list` с параметрами

- Для просмотра определённых строк не основного файла используйте `list` с параметрами
  - Установите точку останова в файле `calculate.c` на строке номер 21
  - Выведите информацию об имеющихся в проекте точка останова
  - Запустите программу внутри отладчика и убедитесь, что программа останавливается в момент прохождения точки останова
  - Отладчик выдаст информацию, а команда `backtrace` покажет весь стек вызываемых функций от начала программы до текущего места
  - Посмотрите, чему равно на этом этапе значение переменной `Numeral`. На экран должно быть выведено число 5
  - Сравните с результатом вывода на экран после использования команды
  - Уберите точки останова
7. С помощью утилиты `splint` попробуйте проанализировать коды файлов `calculate.c` и `main.c`.

### 3 Выполнение лабораторной работы

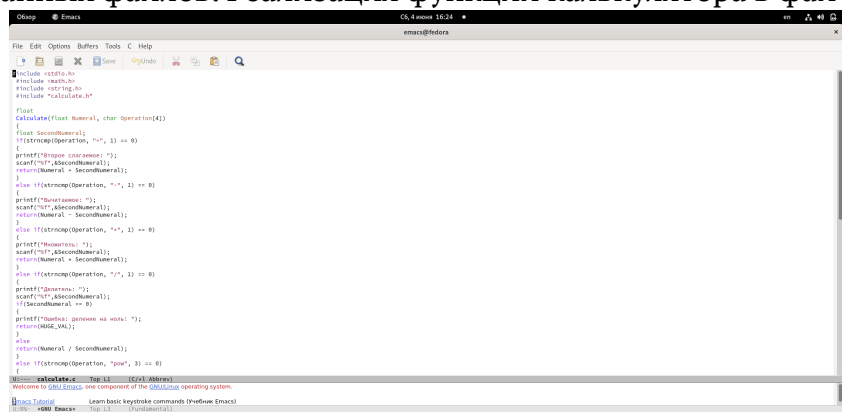
1. Создаём необходимые каталоги, в нем нужные файлы. (рис. ??)



```
edistar@kayafedora: ~/homework/study/2022/ОперационныеСистемы/os-intro/lab13/lab_prog
edistar@kayafedora ~$ cd ~/homework/study/2022/ОперационныеСистемы/os-intro/lab13/
edistar@kayafedora lab13$ mkdir lab_prog
edistar@kayafedora lab13$ cd lab_prog
edistar@kayafedora lab_prog$ touch *.h *.c
edistar@kayafedora lab_prog$ ls
calculate.h  main.c
edistar@kayafedora lab_prog$
```

2. Это будет прими-

тивнейший калькулятор, способный складывать, вычитать, умножать и делить, возводить число в степень, брать квадратный корень, вычислять  $\sin$ ,  $\cos$ ,  $\tan$ . При запуске он будет запрашивать первое число, операцию, второе число. После этого программа выведет результат и остановится. Открыв редактор Emacs, приступил к редактированию созданных файлов. Реализация функций калькулятора в фай-

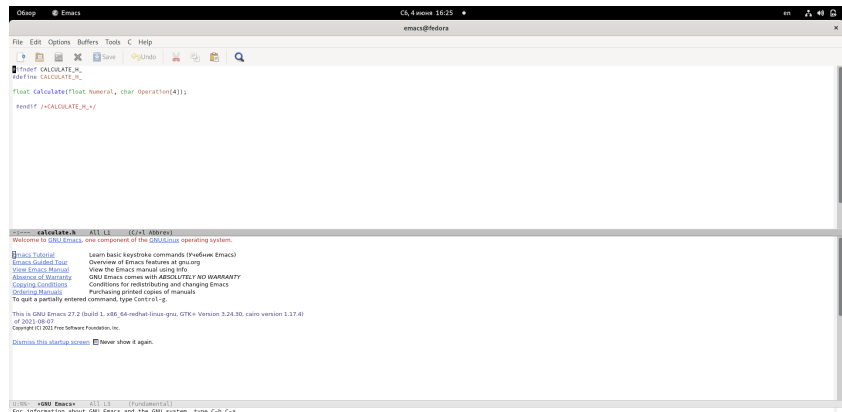


```
edistar@kayafedora ~$ cd ~/homework/study/2022/ОперационныеСистемы/os-intro/lab13/lab_prog
edistar@kayafedora lab13$ cd lab_prog
edistar@kayafedora lab_prog$ touch *.h *.c
edistar@kayafedora lab_prog$ ls
calculate.h  main.c
edistar@kayafedora lab_prog$
```

ле calculate.c (рис. ??)

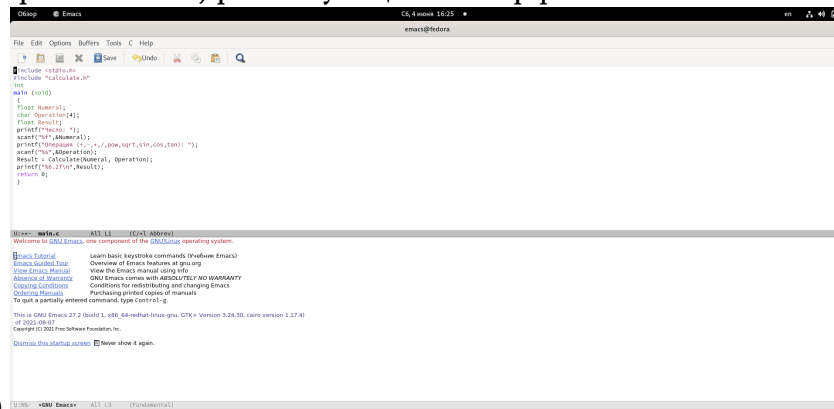
Интерфейсный файл calculate.h, описывающий формат вызова функции





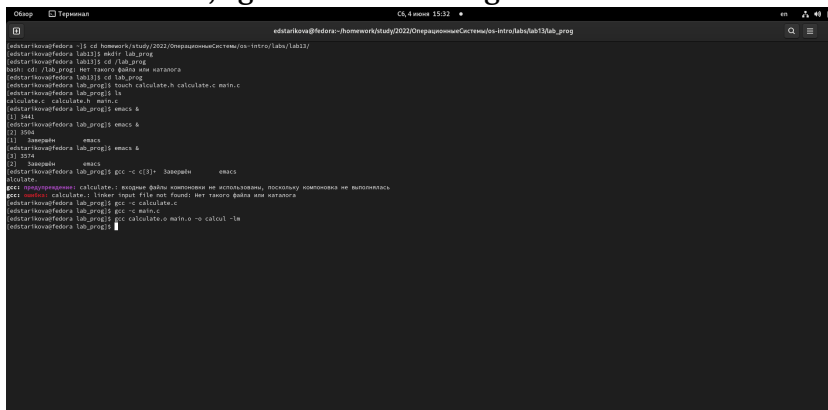
калькулятора (рис. ??)

Основной файл main.c, реализующий интерфейс пользователя к калькулято-



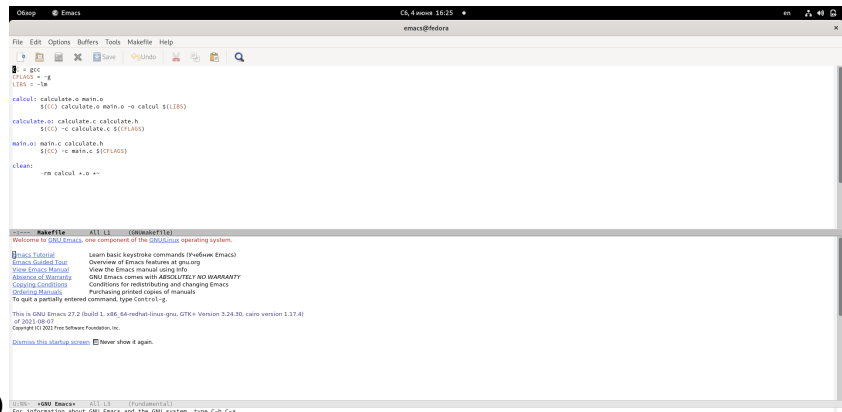
ру(рис. ??)

3. Выполнил компиляцию программы посредством gcc, используя команды «gcc -c calculate.c», «gcc -c main.c» и «gcc calculate.o main.o -o calcul -lm»(рис.



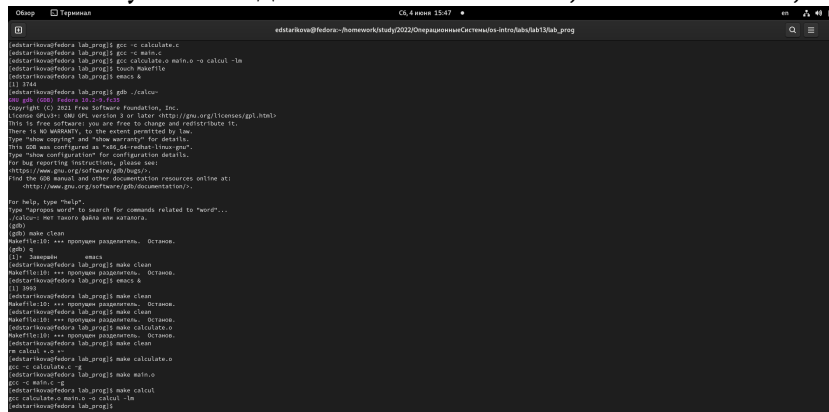
??)

Редактируем

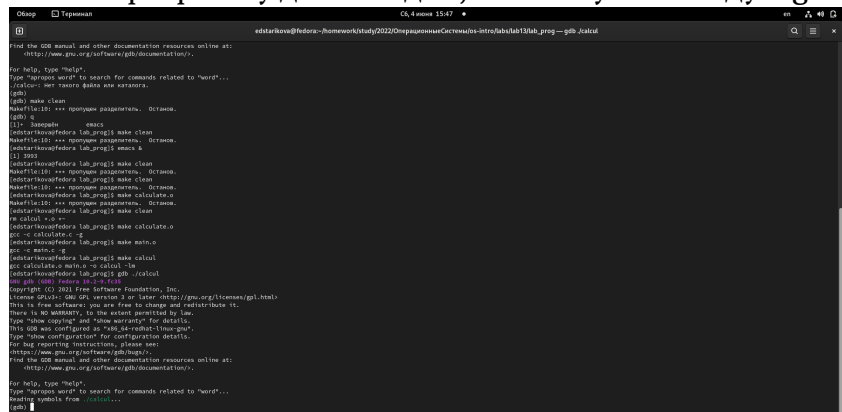


файл Make(рис. ??)

- После этого я удалила исполняемые и объектные файлы из каталога с помощью команды «make clean». Выполнила компиляцию файлов, используя команды «make calculate.o», «make main.o», «male calcul» (рис. ??)

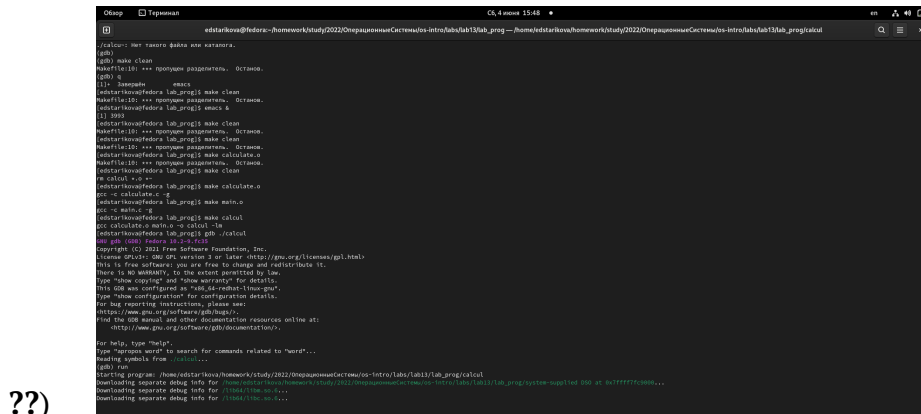


- Далее с помощью gdb выполнил отладку программы calcul. Запустил отладчик GDB, загрузив в него программу для отладки, используя команду: «gdb



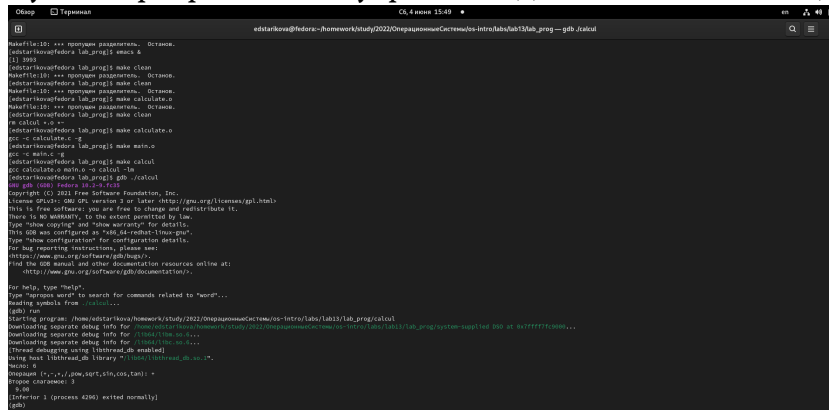
./calcul»(рис. ??)

Для запуска программы внутри отладчика ввёл команду «run» (рис.



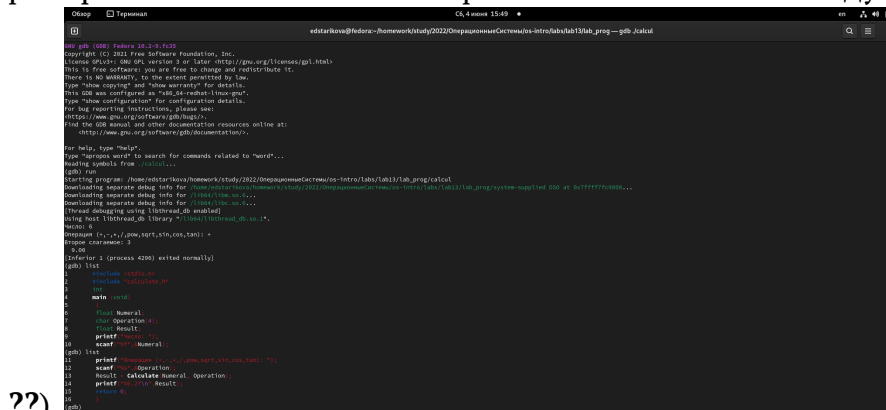
Для за-

пуска программы внутри отладчика ввёл команду «run» (рис. ??)



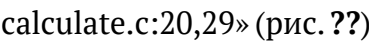
Для просмотра

ра строк с 12 по 15 основного файла использовал команду «list 12,15» (рис.



Для просмотр-

ра определённых строк не основного файла использовал команду «list



```

bash
[0]
[1]
[2]
[3]
[4]
[5]
[6]
[7]
[8]
[9]
[10]
[11]
[12]
[13]
[14]
[15]
[16]
[17]
[18]
[19]
[20]
[21]
[22]
[23]
[24]
[25]
[26]
[27]
[28]
[29]
[30]
[31]
[32]
[33]
[34]
[35]
[36]
[37]
[38]
[39]
[40]
[41]
[42]
[43]
[44]
[45]
[46]
[47]
[48]
[49]
[50]
[51]
[52]
[53]
[54]
[55]
[56]
[57]
[58]
[59]
[60]
[61]
[62]
[63]
[64]
[65]
[66]
[67]
[68]
[69]
[70]
[71]
[72]
[73]
[74]
[75]
[76]
[77]
[78]
[79]
[80]
[81]
[82]
[83]
[84]
[85]
[86]
[87]
[88]
[89]
[90]
[91]
[92]
[93]
[94]
[95]
[96]
[97]
[98]
[99]
[100]
[101]
[102]
[103]
[104]
[105]
[106]
[107]
[108]
[109]
[110]
[111]
[112]
[113]
[114]
[115]
[116]
[117]
[118]
[119]
[120]
[121]
[122]
[123]
[124]
[125]
[126]
[127]
[128]
[129]
[130]
[131]
[132]
[133]
[134]
[135]
[136]
[137]
[138]
[139]
[140]
[141]
[142]
[143]
[144]
[145]
[146]
[147]
[148]
[149]
[150]
[151]
[152]
[153]
[154]
[155]
[156]
[157]
[158]
[159]
[160]
[161]
[162]
[163]
[164]
[165]
[166]
[167]
[168]
[169]
[170]
[171]
[172]
[173]
[174]
[175]
[176]
[177]
[178]
[179]
[180]
[181]
[182]
[183]
[184]
[185]
[186]
[187]
[188]
[189]
[190]
[191]
[192]
[193]
[194]
[195]
[196]
[197]
[198]
[199]
[200]
[201]
[202]
[203]
[204]
[205]
[206]
[207]
[208]
[209]
[210]
[211]
[212]
[213]
[214]
[215]
[216]
[217]
[218]
[219]
[220]
[221]
[222]
[223]
[224]
[225]
[226]
[227]
[228]
[229]
[230]
[231]
[232]
[233]
[234]
[235]
[236]
[237]
[238]
[239]
[240]
[241]
[242]
[243]
[244]
[245]
[246]
[247]
[248]
[249]
[250]
[251]
[252]
[253]
[254]
[255]
[256]
[257]
[258]
[259]
[260]
[261]
[262]
[263]
[264]
[265]
[266]
[267]
[268]
[269]
[270]
[271]
[272]
[273]
[274]
[275]
[276]
[277]
[278]
[279]
[280]
[281]
[282]
[283]
[284]
[285]
[286]
[287]
[288]
[289]
[290]
[291]
[292]
[293]
[294]
[295]
[296]
[297]
[298]
[299]
[300]
[301]
[302]
[303]
[304]
[305]
[306]
[307]
[308]
[309]
[310]
[311]
[312]
[313]
[314]
[315]
[316]
[317]
[318]
[319]
[320]
[321]
[322]
[323]
[324]
[325]
[326]
[327]
[328]
[329]
[330]
[331]
[332]
[333]
[334]
[335]
[336]
[337]
[338]
[339]
[340]
[341]
[342]
[343]
[344]
[345]
[346]
[347]
[348]
[349]
[350]
[351]
[352]
[353]
[354]
[355]
[356]
[357]
[358]
[359]
[360]
[361]
[362]
[363]
[364]
[365]
[366]
[367]
[368]
[369]
[370]
[371]
[372]
[373]
[374]
[375]
[376]
[377]
[378]
[379]
[380]
[381]
[382]
[383]
[384]
[385]
[386]
[387]
[388]
[389]
[390]
[391]
[392]
[393]
[394]
[395]
[396]
[397]
[398]
[399]
[400]
[401]
[402]
[403]
[404]
[405]
[406]
[407]
[408]
[409]
[410]
[411]
[412]
[413]
[414]
[415]
[416]
[417]
[418]
[419]
[420]
[421]
[422]
[423]
[424]
[425]
[426]
[427]
[428]
[429]
[430]
[431]
[432]
[433]
[434]
[435]
[436]
[437]
[438]
[439]
[440]
[441]
[442]
[443]
[444]
[445]
[446]
[447]
[448]
[449]
[450]
[451]
[452]
[453]
[454]
[455]
[456]
[457]
[458]
[459]
[460]
[461]
[462]
[463]
[464]
[465]
[466]
[467]
[468]
[469]
[470]
[471]
[472]
[473]
[474]
[475]
[476]
[477]
[478]
[479]
[480]
[481]
[482]
[483]
[484]
[485]
[486]
[487]
[488]
[489]
[490]
[491]
[492]
[493]
[494]
[495]
[496]
[497]
[498]
[499]
[500]
[501]
[502]
[503]
[504]
[505]
[506]
[507]
[508]
[509]
[510]
[511]
[512]
[513]
[514]
[515]
[516]
[517]
[518]
[519]
[520]
[521]
[522]
[523]
[524]
[525]
[526]
[527]
[528]
[529]
[530]
[531]
[532]
[533]
[534]
[535]
[536]
[537]
[538]
[539]
[540]
[541]
[542]
[543]
[544]
[545]
[546]
[547]
[548]
[549]
[550]
[551]
[552]
[553]
[554]
[555]
[556]
[557]
[558]
[559]
[560]
[561]
[562]
[563]
[564]
[565]
[566]
[567]
[568]
[569]
[570]
[571]
[572]
[573]
[574]
[575]
[576]
[577]
[578]
[579]
[580]
[581]
[582]
[583]
[584]
[585]
[586]
[587]
[588]
[589]
[590]
[591]
[592]
[593]
[594]
[595]
[596]
[597]
[598]
[599]
[600]
[601]
[602]
[603]
[604]
[605]
[606]
[607]
[608]
[609]
[610]
[611]
[612]
[613]
[614]
[615]
[616]
[617]
[618]
[619]
[620]
[621]
[622]
[623]
[624]
[625]
[626]
[627]
[628]
[629]
[630]
[631]
[632]
[633]
[634]
[635]
[636]
[637]
[638]
[639]
[640]
[641]
[642]
[643]
[644]
[645]
[646]
[647]
[648]
[649]
[650]
[651]
[652]
[653]
[654]
[655]
[656]
[657]
[658]
[659]
[660]
[661]
[662]
[663]
[664]
[665]
[666]
[667]
[668]
[669]
[670]
[671]
[672]
[673]
[674]
[675]
[676]
[677]
[678]
[679]
[680]
[681]
[682]
[683]
[684]
[685]
[686]
[687]
[688]
[689]
[690]
[691]
[692]
[693]
[694]
[695]
[696]
[697]
[698]
[699]
[
```

ду «list calculate.c:15,22» (рис. ??)

## 4 Выводы

В ходе выполнения данной лабораторной работы я приобрела простейшие навыки разработки, анализа, тестирования и отладки приложений в ОС типа UNIX/Linux на примере создания на языке программирования С калькулятора с простейшими функциями.