

ЛАБОРАТОРНАЯ РАБОТА №5

Отчёт

Глобин Никита Анатольевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Открыть и воспользоваться Midnight Commander	6
2.2	Создание и работа с файлом через Midnight Commander	7
2.3	Запуск полученной программы	10
2.4	Подключение внешнего файла in_out.asm и создание копии файла через Midnight Commander	11
2.5	Самостоятельная работа	12
3	Выводы	16
	Список литературы	17

Список иллюстраций

2.1	photo 1	6
2.2	photo 2	7
2.3	photo 3	8
2.4	photo 4	9
2.5	photo 5	10
2.6	photo 6	10
2.7	photo 7	11
2.8	photo 8	11
2.9	photo 9	12
2.10	photo 10	12
2.11	photo 11	12
2.12	photo 12	13
2.13	photo 13	13
2.14	photo 14	14
2.15	photo 15	14
2.16	photo 16	15

Список таблиц

1 Цель работы

Приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера mov и int. # Задание

Открыть и воспользоваться Midnight Commander

Создание и работа с файлом через Midnight Commander

Запуск полученной программы

Подключение внешнего файла in_out.asm и создание копии файла через Midnight Commander

Самостоятельная работа

2 Выполнение лабораторной работы

2.1 Открыть и воспользоваться Midnight Commander

1. Открываем терминал и вводим команду mc (рис. 2.1)

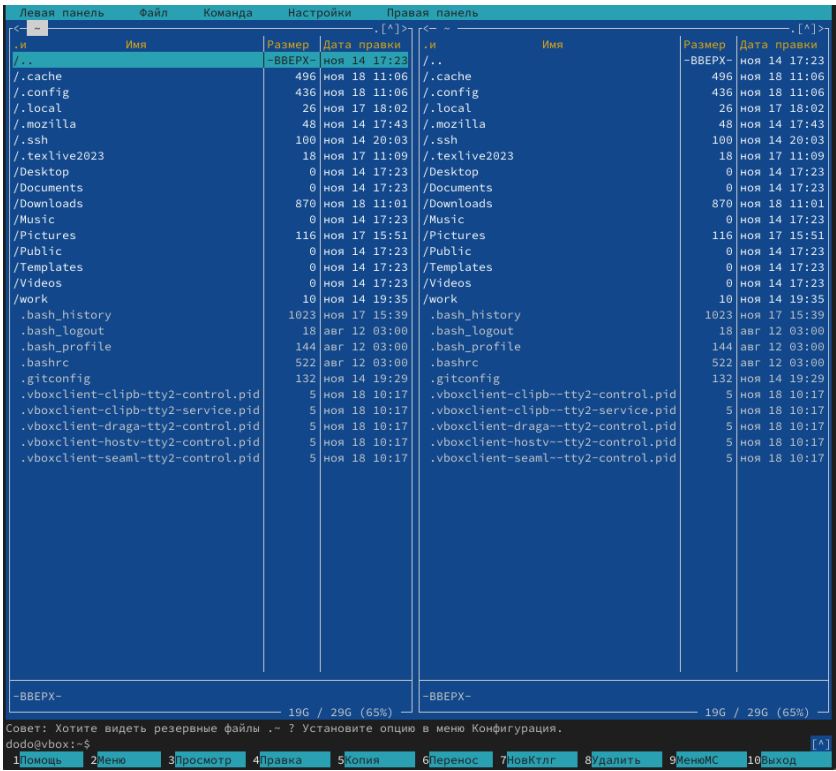


Рис. 2.1: photo 1

2. Используя функционал Midnight Commander, переходим в папку lab05 (рис. 2.2)

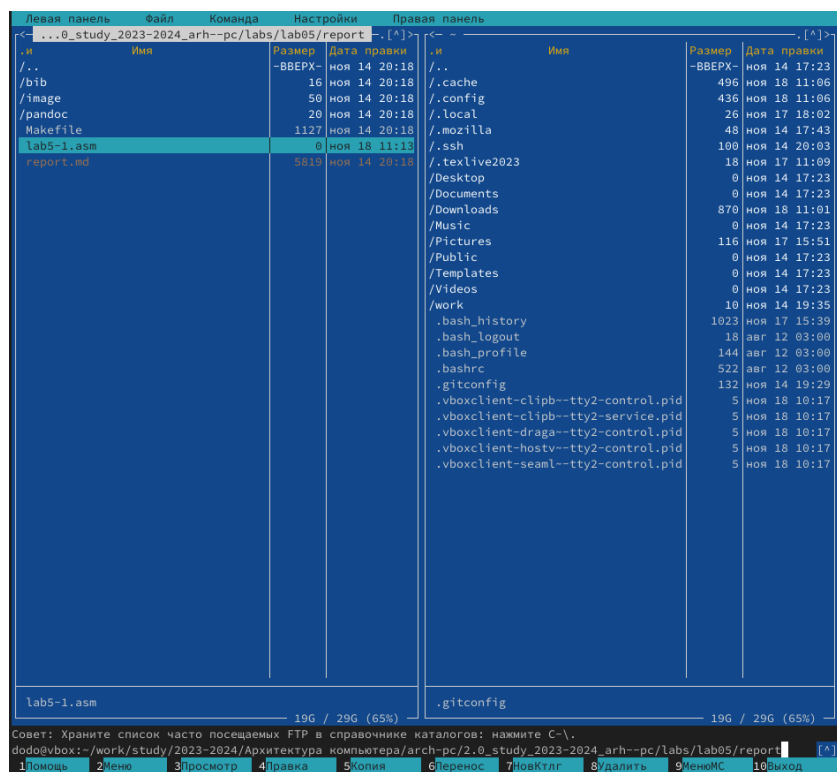


Рис. 2.2: photo 2

2.2 Создание и работа с файлом через Midnight Commander

1. Создаём файл и lab5-1 (рис. 2.3)

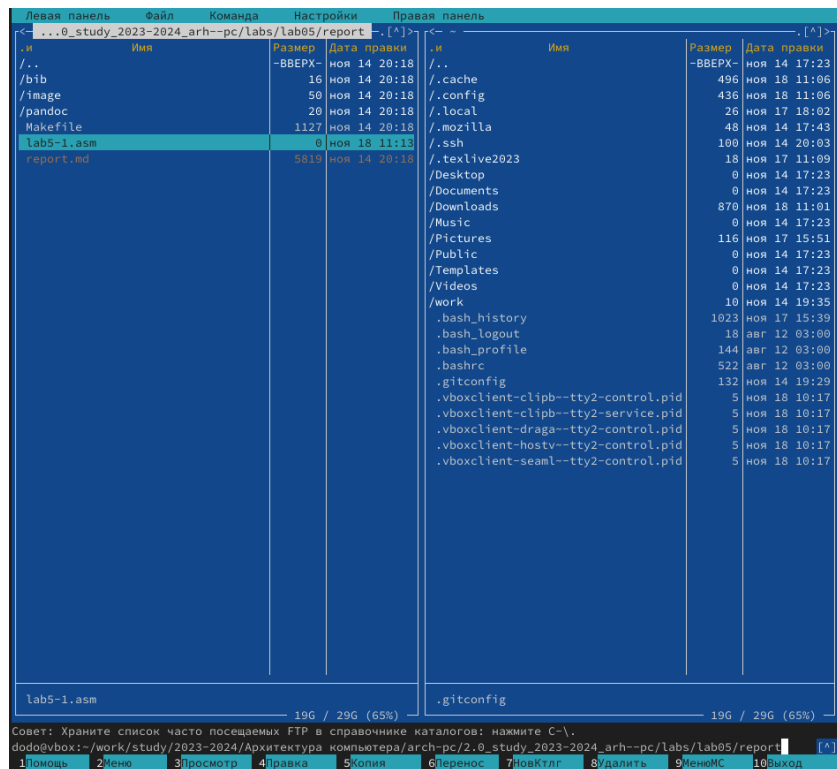
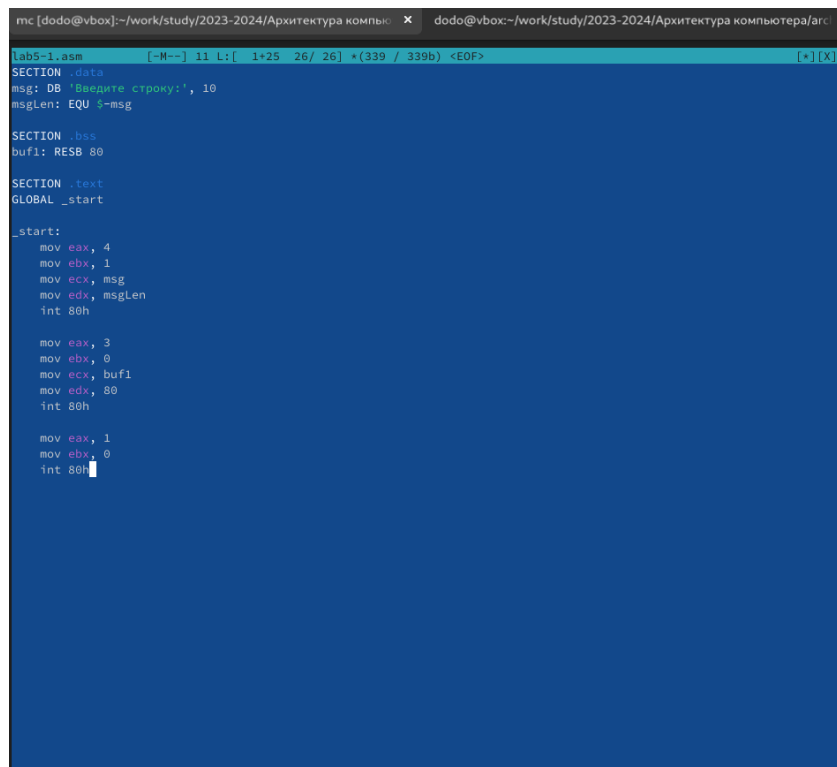


Рис. 2.3: photo 3

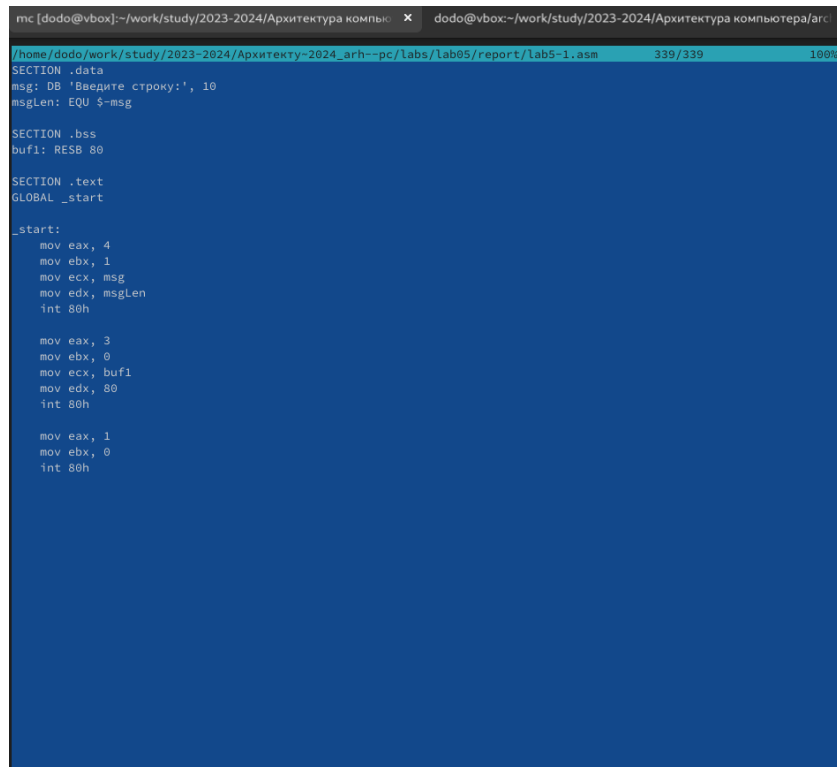
2. записываем в него программу (рис. 2.4)



```
mc [dodo@vbox:~/work/study/2023-2024/Архитектура компык: x dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arc  
lab5-1.asm [-M--] 11 L: [ 1+25 26/ 26] *(339 / 339b) <EOF> [*] [X]  
SECTION .data  
msg: DB 'Введите строку:', 10  
msgLen: EQU $-msg  
  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
GLOBAL _start  
  
_start:  
    mov eax, 4  
    mov ebx, 1  
    mov ecx, msg  
    mov edx, msgLen  
    int 80h  
  
    mov eax, 3  
    mov ebx, 0  
    mov ecx, buf1  
    mov edx, 80  
    int 80h  
  
    mov eax, 1  
    mov ebx, 0  
    int 80h
```

Рис. 2.4: photo 4

3. Сохраняем программу и выходим. Затем через клавишу F3 проверяем схранился ли файл (рис. 2.5)

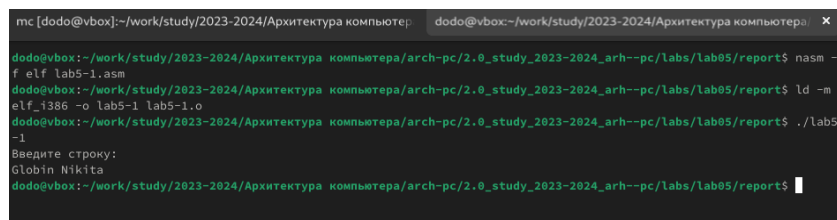


```
mc [dodo@vbox]:~/work/study/2023-2024/Архитектура компь: x dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-  
/home/dodo/work/study/2023-2024/Архитектура-2024_arh--pc/labs/Lab05/report/lab5-1.asm 339/339 100%  
SECTION .data  
msg: DB 'Введите строку:', 10  
msgLen: EQU $-msg  
  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
GLOBAL _start  
  
_start:  
    mov eax, 4  
    mov ebx, 1  
    mov ecx, msg  
    mov edx, msgLen  
    int 80h  
  
    mov eax, 3  
    mov ebx, 0  
    mov ecx, buf1  
    mov edx, 80  
    int 80h  
  
    mov eax, 1  
    mov ebx, 0  
    int 80h
```

Рис. 2.5: photo 5

2.3 Запуск полученной программы

1. компилируем полученный файл и вводим туда свою фамилию и имя (рис. 2.6)



```
mc [dodo@vbox]:~/work/study/2023-2024/Архитектура компьют: dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch- x  
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ nasm -  
f elf lab5-1.asm  
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ld -m  
elf_i386 -o lab5-1 lab5-1.o  
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ./lab5  
-1  
Введите строку:  
Globin Nikita  
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$
```

Рис. 2.6: photo 6

2.4 Подключение внешнего файла in_out.asm и создание копии файла через Midnight Commander

1. скачиваем и переносим файл in_out.asm в нашу папку с файлом (рис. 2.7)

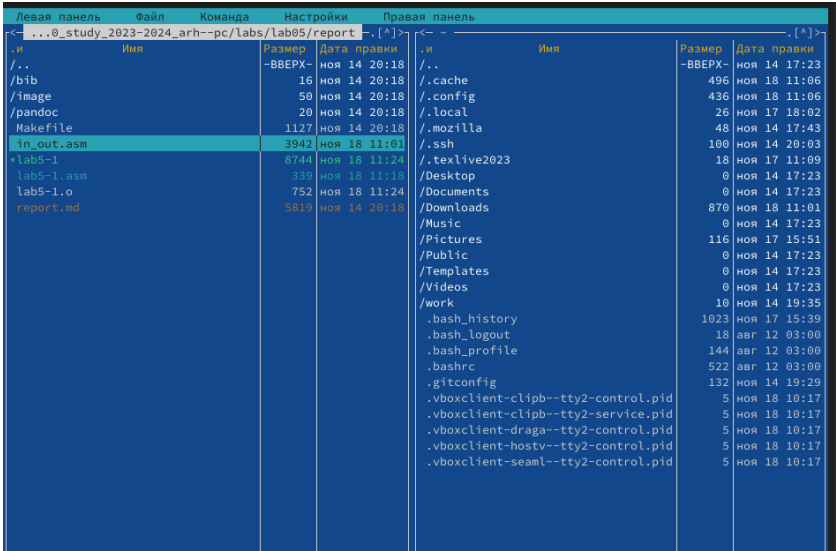


Рис. 2.7: photo 7

2. С помощью функциональной клавиши F6 создайте копию файла lab5-1.asm с именем lab5-2.asm. (рис. 2.8)

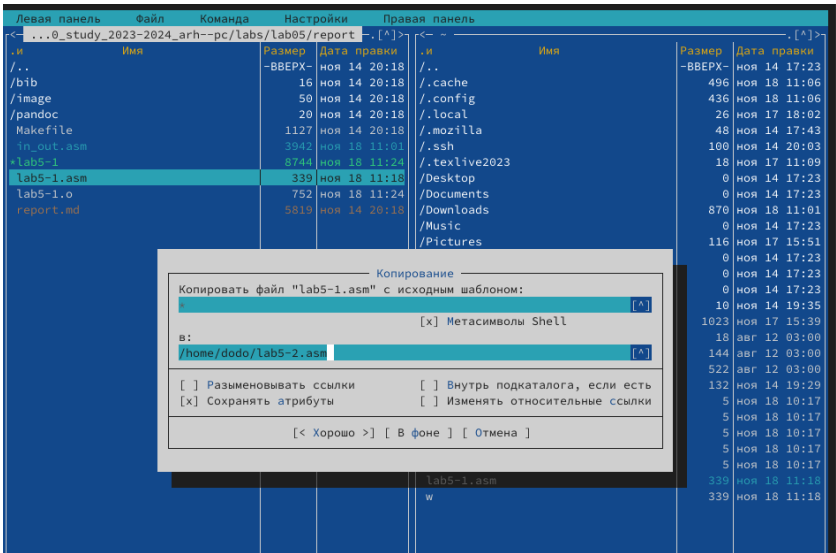
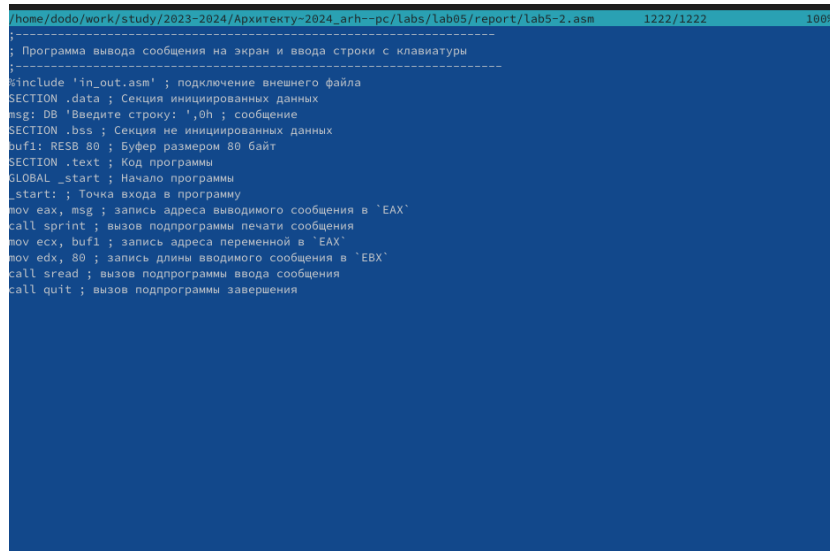


Рис. 2.8: photo 8

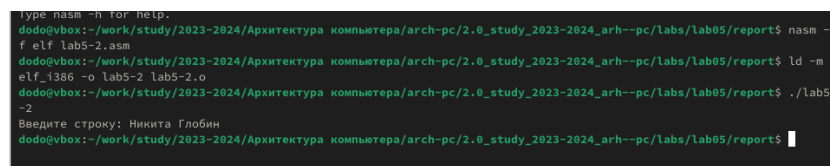
- Исправляем текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm (рис. 2.9)



```
home/dodo/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report/lab5-2.asm 1222/1222 100%
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения
```

Рис. 2.9: photo 9

- компилируем и запускаем эту программу (рис. ??)



```
Type nasm -h for help.
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ nasm -f elf lab5-2.asm
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ld -m elf_i386 -o lab5-2 lab5-2.o
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ./lab5-2
Введите строку: Никита Глобин
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$
```

Рис. 2.10: photo 10

2.5 Самостоятельная работа

- Создаём копию файла lab5-1.asm и пишем программу которая будет выполнять (рис. ??) (рис. ??) : • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.

photo 11

Рис. 2.11: photo 11

```

;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80 байт
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра
;----- Системный вызов 'write'
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'buf1' длиной 80
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,buf1 ; Адрес строки 'buf1' в 'ecx'
mov edx,80 ; Размер строки 'buf1' в 'edx'
int 80h ; Вызов ядра
;----- Системный вызов 'exit' -----
; После вызова инструкции 'int 80h' программа завершит работу
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра

```

Рис. 2.12: photo 12

2. запускаем программу (рис. ??)

```

Введите строку: Никита Глобин
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ nasm -
f elf lab5-1-1.asm
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ld -m
elf_i386 -o lab5-1-1 lab5-1-1.o
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$ ./lab5
-1-1
Введите строку:
Глобин
Глобин
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh--pc/labs/lab05/report$

```

Рис. 2.13: photo 13

3. Создаём копию файла lab5-2.asm и пишем программу которая будет выпол-

нять (рис. ??) (рис. ??) : • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.

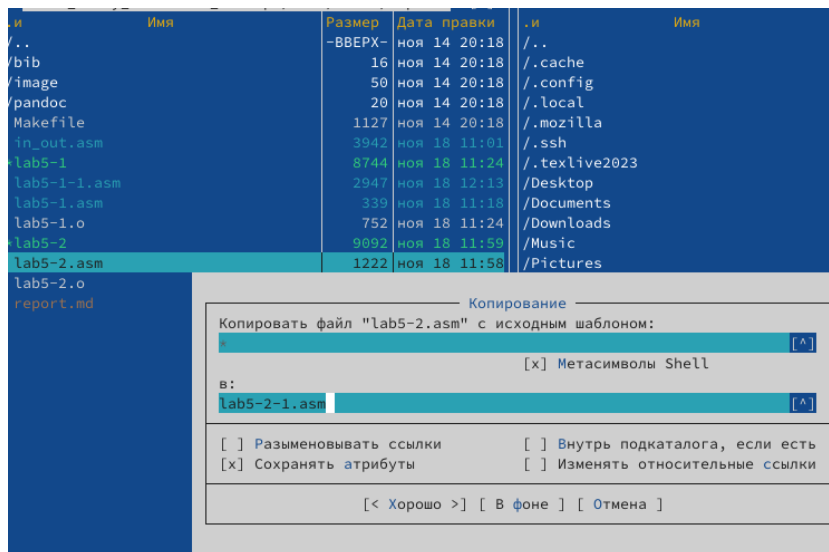


Рис. 2.14: photo 14

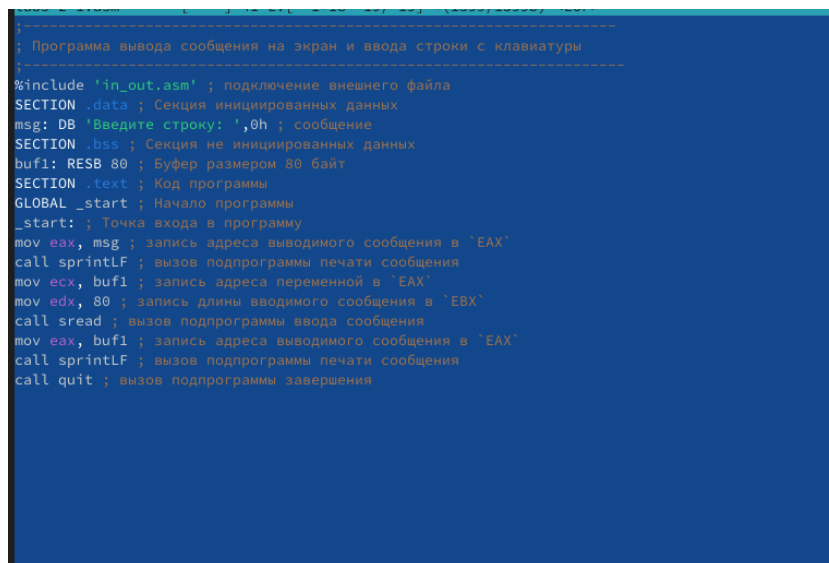


Рис. 2.15: photo 15

4. запускаем программу (рис. ??)

```
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh---pc/labs/lab05/report$ nasm -f elf lab5-2-1.asm
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh---pc/labs/lab05/report$ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh---pc/labs/lab05/report$ ./lab5-2-1
Введите строку:
Глобин
Глобин
dodo@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/2.0_study_2023-2024_arh---pc/labs/lab05/report$
```

Рис. 2.16: photo 16

3 Выводы

В ходе лабораторной работы я приобрёл навык работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`. Теперь я могу писать простейшие программы на `asm` и использовать Midnight Commander, что в свою очередь улучшило моё понимание компьютера.

Список литературы