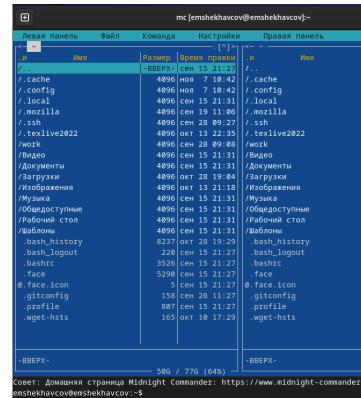
```
N_{\overline{2}}5
1
1.1
1.1.1
1.1.2
1
         1 2
                 13
                                  1 4
                                                         2 4.1
                                                                        mc
2\ 4.2
                             NASM 4 4.3
                                                           64.4
                        12 6
                9 5
1.2 1
                                                     Midnight Commander,
                      mov int. ## 2
                                                       mc 2.
          NASM 3.
1.3 3
Midnight Commander (
                            mc) —
                                      , .. mc
                                                                . Midnight
{\bf Commander}
                                                                    NASM,
                                     (SECTION .text),
                 (SECTION .data)
                                                      ) (SECTION .bss).
                                              DB, DW, DD, DQ DT,
                          .data
                                               : - DB (define byte) —
                ; - DW (define word) —
DD (define double word) —
                                                             ); - DQ (define
                                                    ); - DT (define ten bytes)
quad word) —
                      10
                                      DB
                                                               . mov dst, src
                      mov
                                                                  (register),
         dst —
                      \operatorname{src} —
        (memory)
                                    (const).
                                                                  0-255.
                       . int n
                                    sys_calls n=80h (
           Linux
```

).

1.4 4

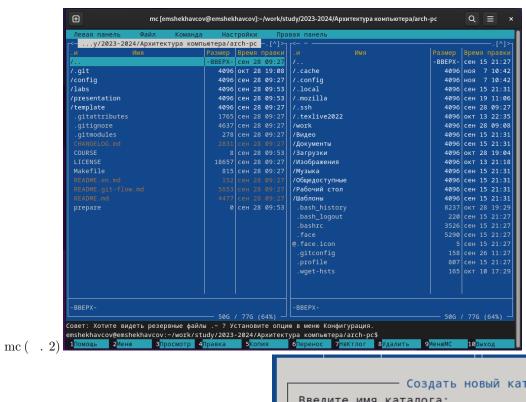
1.4.1 4.1 mc



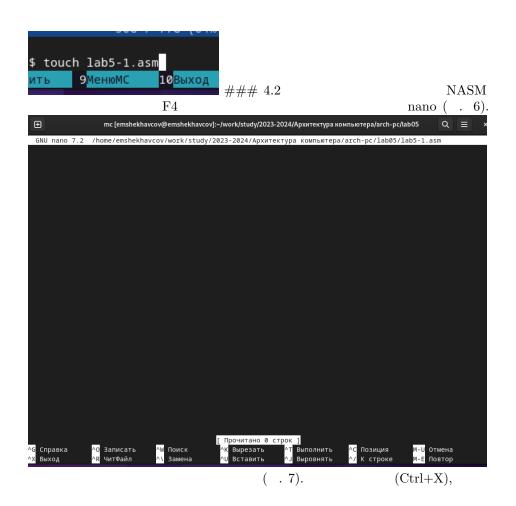
4Правка **5**Копия **6**Перенос **7**НвКтлог **8**Уда

mc (. 1). 1 Помощь 2 Меню Midnight Commander, ~/work/study/2023-2024/

/arch-pc,



— Создать новый каталог Введите имя каталога: lab05 [< Дальше >] [Прервать] F7lab05 (. 3). ⊞ mc [emshekhavcov@emshekhavcov]:~/work/study Файл Левая панель Команда Настройки ...-2024/Архитектура компьютера/arch-pc/lab05 Размер Время правки -BBEPX- ноя 7 10:45 . 4). touch lab5-1.asm, (. 5).



```
Đ
                                            mc [emshekhavcov@emshekhavcov]:~/work/study/2023-2024/Архитектура компьютера/arch-pc/lab05
                                                                                                                                a =
                          GNU nano 7.2 /home/emshekhavcov/work/study/2023-2024/Архитектура компьютера/arch-pc/lab05/lab5-1.asm *
                                     ta : Секция инициированных данных
                                  'Введите строку:',10 ; сообщение плюс
                            символ перевода строки
                                    ı $-msg ; Длина переменной 'msg'
                                  .bss ; Секция не инициированных данных
                                  58 80 ; Буфер размером 80 байт
------ Текст программы -----
                                  .text ; Код программы
                                 _start ; Начало программы
                                 ; Точка входа в программу
                           ------ Системный вызов `write`
После вызова инструкции 'int 80h' на экран будет
выведено сообщение из переменной 'msg' длиной 'ms
                          mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
                          nov ecx,msg ; Адрес строки 'msg' в 'ecx'
nov 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
nov eax, 3 ; Системный вызов для чтения (sys_read)
nov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
                          mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
                           ------ 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
                                         ^О Записать
^В ЧитФайл
                                                                            Вырезать
        (Y, Enter). AX выход
                                                           Замена
                                                                         ^U Вставить
                             F3
                                                                                                             . 8).
        mc [emshekhavcov@emshekhavcov]:~/work/study/2023-2024/Архитектур...
                                                                                                Q ≡
                                                                                                               ×
 /home/emshekhavcov/work~ch-pc/lab05/lab5-1.asm
                                                                           1492/2096
                                                                                                             71<mark>%</mark>
SECTION .data ; Секция инициированных данных
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициированных данных
buf1: RESB 80 ; Буфер размером 80 байт
 ;-----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
 ----- Системный вызов `write`
 ; 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` -----
  строки, которая будет записана в переменную 'buf1' размером 80 байт
mov eax, 3 ; Системный вызов для чтения (sys_read)
1<mark>Помощь 2Раз∼рн З</mark>выход <mark>4</mark>Нех 5Пер∼ти б 7
```

nasm -f elf lab5-1.asm.

7Поиск <mark>8</mark>Исх~ый **9**Формат**10**Выход

lab5-

```
ld -m elf_i386 -o lab5-1 lab5-1.o ( . 9).
1.o.
                          lab05$ nasm -f elf lab5-1.asm
                          lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
                 lab5-1.
                                           :"
                                lab05$ ./lab5-1
                                Введите строку:
                                Шехавцов Евгений Михайлович
                      (10).
                                                                       ### 4.3
                         in out.asm
      \oplus
                                                                        Q
                           mc [emshekhavcov@emshekhavcov]:~/Загрузки
                                                                            Команда
                                            Настройки
         ~/Загрузки
                                                               -BBEPX-
                                                                      сен 15 21:27
                                                                  4096 ноя 7 10:42
                          3942 ноя 7 11:10
                                             /.config
       in_out.asm
                                                                  4096 ноя 7 10:42
      install-~.tar.gz
msg51091~546.jpg
msg51091~561.jpg
                                                                  4096 сен 15 21:31
                                             /.local
                                                                  4096 сен 19 11:06
                                                                  4096 сен 28 09:27
11).
                     F5
                                                                       lab05 (
                                in out.asm
                                   — Копирование –
        Копировать файл "in_out.asm" с исходным шаблоном:
                                            [x] Метасимволы shell
        [ ] Разыменовывать ссылки
                                            [ ] Внутрь подкаталога, если есть
        [х] Сохранять атрибуты
                                            [ ] Изменять относительные ссылки
                        [< Дальше >] [ В фоне ] [ Прервать ]
12).
                    F5
                                lab5-1
```

```
— Копирование -
                                  Копировать файл "lab5-1.asm" с исходным шаблоном:
                                                                      [x] Метасимволы shell
                                  пьютера/arch-pc/lab05/lab5-2.asm
                                  [ ] Разыменовывать ссылки
                                                                      [ ] Внутрь подкаталога, если есть
                                  [х] Сохранять атрибуты
                                                                      [ ] Изменять относительные ссылки
                                                  [< Дальше >] [ В фоне ] [ Прервать ]
                      ( . 13).
mc
                lab5-2.asm
                                           nano ( . 14),
                                      mc [emshekhavcov@emshekhavcov]:~/work/study/2023-2024/Архитектур...
                                 .../work/study/2023-2024/Архитектура компьютера/arch-pc/lab05/lab5-2.asm *
                                 %include 'in_out.asm' ; подключение внешнего файла
                                        .data ; Секция инициированных данных
                                        'Введите строку: ',0h ; сообщение
                                        .bss ; Секция не инициированных данных
                                           80 ; Буфер размером 80 байт
                                        .text ; Код программы
                                       _start ; Начало программы
                                        ; Точка входа в программу
                                 mov eax, msg ; запись адреса выводимого сообщения в `EAX`
                                 call sprintLF ; вызов подпрограммы печати сообщения
                                mov ecx, buf1 ; запись адреса переменной в
                                mov edx, 80 ; запись длины вводимого сообщения в `EBX`
                                call sread ; вызов подпрограммы ввода сообщения
                                call quit ; вызов подпрограммы завершения
                   in out.asm.
                                             nasm -f elf lab5-2.asm.
           lab5-2.o.
                                                               ld -m \ elf\_i386
-o lab5-2 lab5-2.o
                                       lab5-2.
                                                                           15).
                                                       lab05$ ./lab5-2
 lab05$ nasm -f elf lab5-2.asm
                                                       Введите строку:
                                                       Шехавцов Евгений Михайлович
 lab05$ ld -m elf_i386 -o lab5-2 lab5-2.o
         lab5-2.asm
                                                         F4.
                                   nano
sprintLF sprint.
                                                                       ( . 16).
```

```
.../work/study/2023-2024/Архитектура компьютера/arch-pc/lab05/lab5-2.asm *
 %include 'in_out.asm' ; подключение внешнего файла
        .data ; Секция инициированных данных
        'Введите строку: ',0h ; сообщение
        .bss ; Секция не инициированных данных
        SB 80 ; Буфер размером 80 байт
        .text ; Код программы
       _start ; Начало программы
       ; 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 `
 mov edx, 80 ; запись длины вводимого сообщения в `EBX`
 call sread ; вызов подпрограммы ввода сообщения
 call quit ; вызов подпрограммы завершения
      lab05$ nasm -f elf lab5-2.asm
      lab05$ ld -m elf_i386 -o lab5-2-2 lab5-2.o
      lab05$ ./lab5-2-2
      Введите строку: Шехавцов Евгений Михайлович
17).
            lab5-2
                         lab5-2-2
                         sprintLF sprint. ### 4.4
                                                                F5 ( . 18).
    1.
                lab5-1.asm
                               lab5-1-1.asm
                               Копирование -
  Копировать файл "lab5-1.asm" с исходным шаблоном:
                                                                     [^]
                                      [x] Метасимволы shell
   пьютера/arch-pc/lab05/lab5-1-1.asm
                                [ ] Внутрь подкаталога, если есть [ ] Изменять относительные ссылки
   [ ] Разыменовывать ссылки
   [х] Сохранять атрибуты
                  [< Дальше >] [ В фоне ] [ Прервать ]
```

8

F4

```
19).
      mc [emshekhavcov@emshekhavcov]:~/work/study/2023-2024/Архитектур...
  ..ork/study/2023-2024/Архитектура компьютера/arch-pc/lab05/lab5-1-1.asm
        .data ; Секция инициированных данных
        'Введите строку:',10
          U $-msg ; Длина переменной 'msg'
        .bss ; Секция не инициированных данных
          80 ; Буфер размером 80 байт
         text ; Код программы
       _start ; Начало программы
       ; Точка входа в программу
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла '1' - стандартный вывод
mov ecx,buf1 ; Адрес строки buf1 в есх
mov edx,buf1 ; Размер строки buf1
int 80h ; Вызов ядра
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
   Справка
             ^0 Записать
                         ^₩ Поиск
                                        Вырезать
                                                   ^Т Выполнить ^С Позиция
   Выход
               ЧитФайл
                            Замена
                                         Вставить
                                                     Выровнять
                                                                  К строке
2.
                                                                      lab5-1-1,
                lab5-1-1.o,
                   lab05$ nasm -f elf lab5-1-1.asm
                   lab05$ ld -m elf_i386 -o lab5-1-1 lab5-1-1.o
                   lab05$ ./lab5-1-1
                   Введите строку:
                   Шехавцов Евгений Михайлович
        (.20).
             1:
SECTION .data;
                                      msg: DB '
                                                           :',10 msgLen: EQU
                   'msg' SECTION .bss;
$-msg;
                                                              buf1: RESB 80;
               SECTION .text;
                                          GLOBAL _start;
                                                                        \_start:
                 mov eax, 4;
                                                (sys write) mov ebx,1;
                                           'msg' 'ecx' mov edx,msgLen;
   1 -
                   mov ecx,msg;
    'msg' 'edx' int 80h;
                                 mov eax, 3;
                                                               (sys_read) mov
ebx, 0;
                  0 -
                                 mov ecx, buf1;
                                                                      mov edx,
```

```
80;
                   int 80h;
                                    mov eax, 4;
                                                                  (sys_write)
mov ebx,1;
                      '1' -
                                                               buf1 ecx mov
                                       mov ecx, buf1;
edx,buf1:
                   buf1 int 80h;
                                       mov eax, 1:
                                                                    (sys exit)
                          0 (
mov ebx,0;
                                    ) int 80h;
3.
                 lab5-2.asm
                                      lab5-2-1.asm
                     *lab5-2-1.asm
                                              9092 ноя
F5 ( .
            21).
                F4
                                                                          22).
                           emshekhavcov@emshekhavcov: ~
  ../work/study/2023-2024/Архитектура компьютера/arch-pc/lab05/lab5-2.asm
 %include 'in_out.asm' ; подключение внешнего файла
        .data ; Секция инициированных данных
        'Введите строку: ',0h ; сообщение
        .bss ; Секция не инициированных данных
         В 80 ; Буфер размером 80 байт
        .text ; Код программы
       _start ; Начало программы
       : ; Точка входа в программу
 mov eax, msg ; запись адреса выводимого сообщения в
 call sprint ; вызов подпрограммы печати сообщения
 mov ecx, buf1 ; запись адреса переменной в
 mov edx, 80 ; запись длины вводимого сообщения в
 call sread ; вызов подпрограммы ввода сообщения
 call quit ; вызов подпрограммы завершения
4.
               lab5-2-1.o,
                                                                lab5-2-1,
                       lab05$ nasm -f elf lab5-2-1.asm
                       lab05$ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
                       lab05$ ./lab5-2-1
                       Введите строку: Шехавцов Евгений Михайлович
             ( . 23).
               3:
%include 'in_out.asm' SECTION .data ;
                                                              msg: DB '
               SECTION .bss;
                                                    buf1: RESB 80;
   :',0h;
      SECTION .text;
80
                                GLOBAL start;
                                                              start:;
       mov eax, msg;
                                              EAX call sprint;
      mov ecx, buf1;
                                      EAX mov edx, 80;
EBX call sread;
                                      mov eax, 4;
                                                                   (sys_write)
                      '1' -
                                      mov ecx, buf1;
                                                            buf1 ecx int 80h
mov ebx, 1;
        call quit;
```

1.5 5

Midnight Commander,

mov int.

1.6 6

1. №5