Отчёт по лабораторной работе №5

дисциплина: Архитектура компьютеров

Маслова Анна Павловна

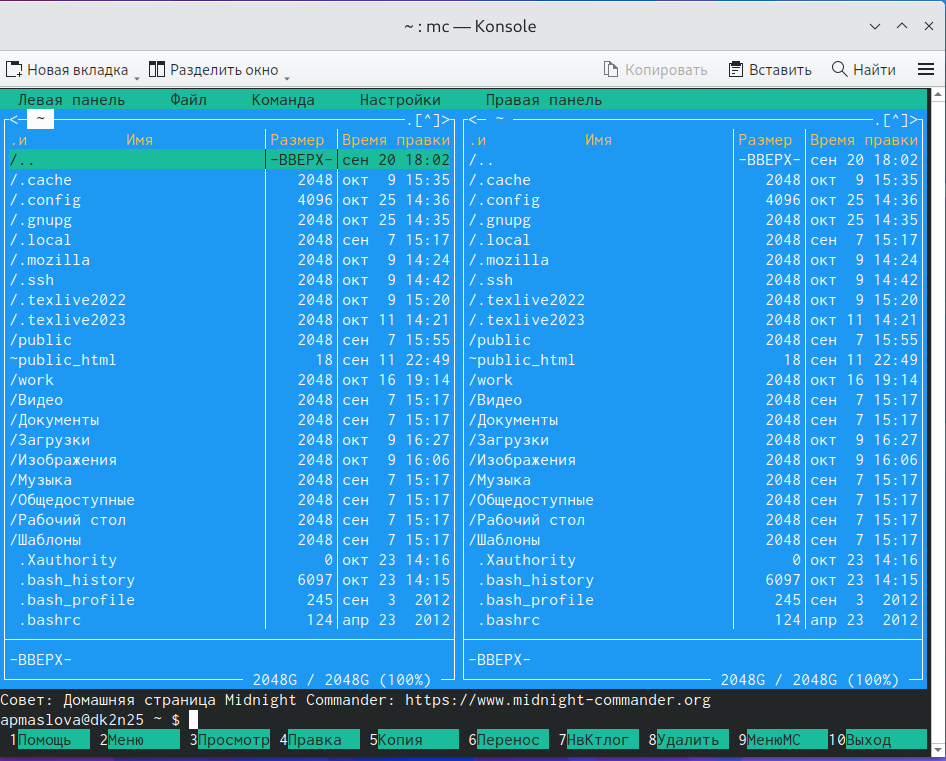
Содержание

# 1 Цель работы

Приобрести навыки работы в *Midnight Commander* на практике. Освоить инструкции языка ассемблера *mov* и *int*.

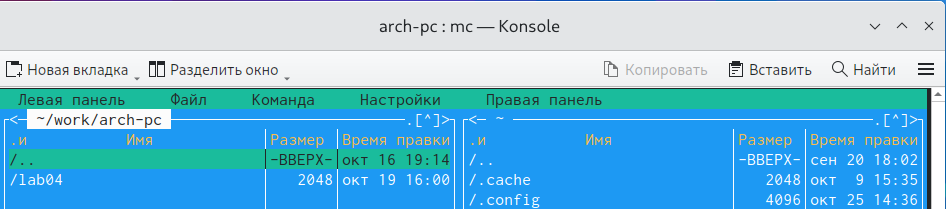
# 2 Выполнение лабораторной работы

С помощью комады mc откроем *Midnight Commander* (рис. ??).



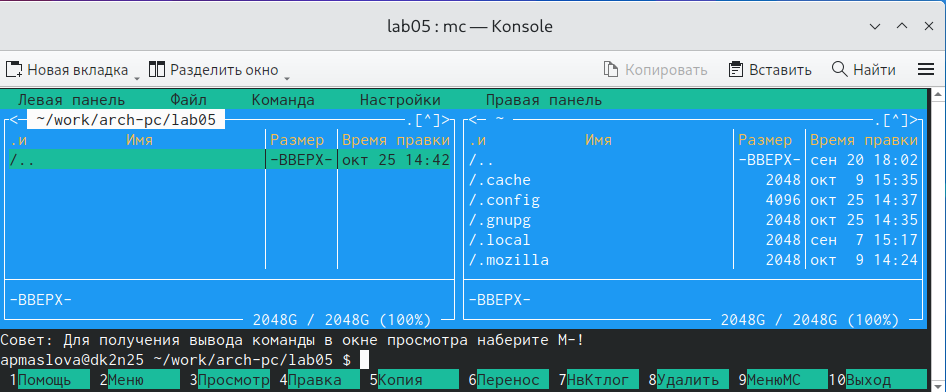
Midnight Commander

Затем с помощью клавиатуры переходим в каталог ~/work/arch-pc (рис. ??).



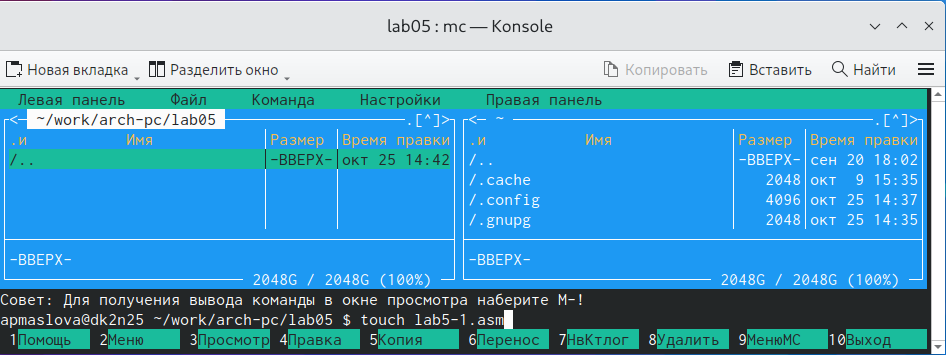
Каталог ~/work/arch-pc

В этом каталоге с помощью клавиши F7 создадим папку *lab05* и перейдём в неё (рис. ??).



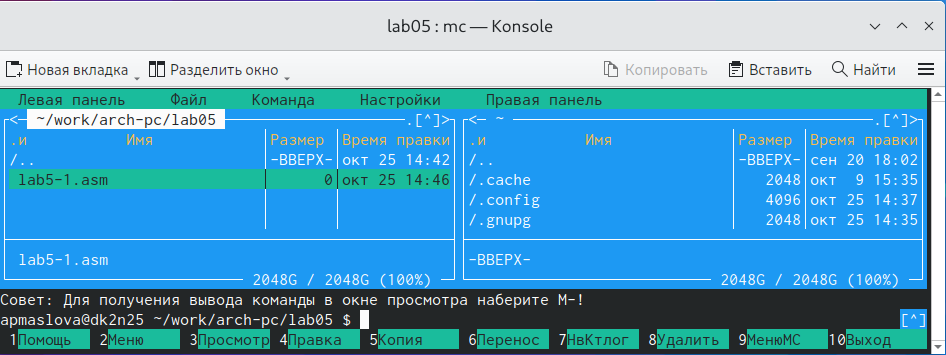
Каталог lab05

Далее ниже строке ввода вводим команду *touch*, с помощью которой создадим файл *lab5-1.asm* (рис. ??).



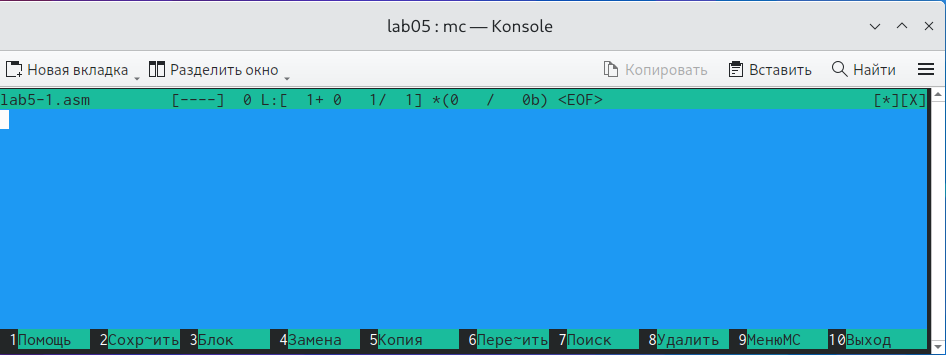
Строка ввода

На рис. ?? видим, что файл создан:



Созданный файл

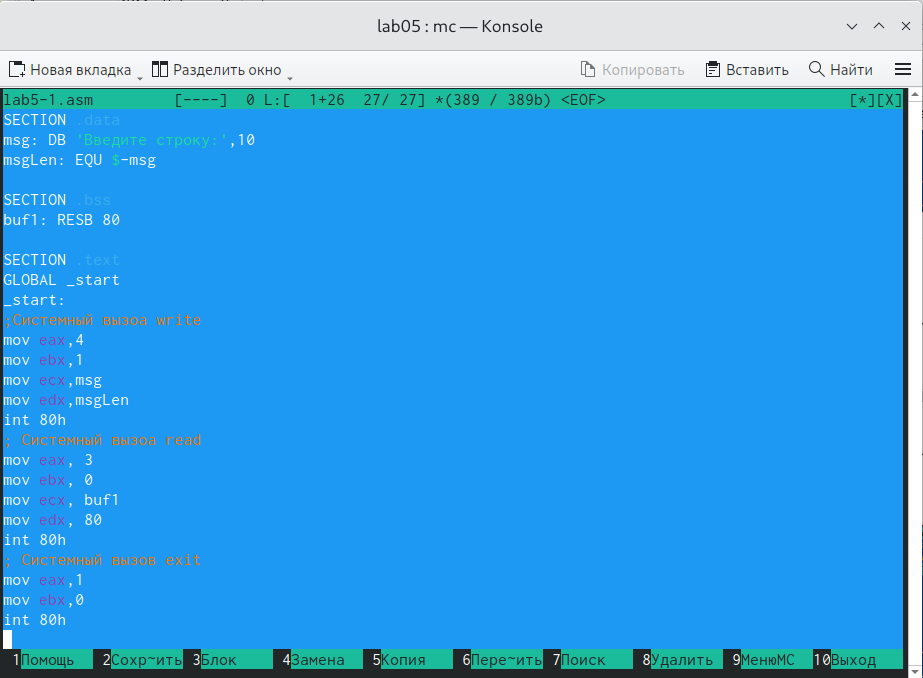
С помощью клавиши F4 откроем созданный файл для редактирования (рис. ??).



Редактор mcedit

Видим, что открылся редактор mcedit.

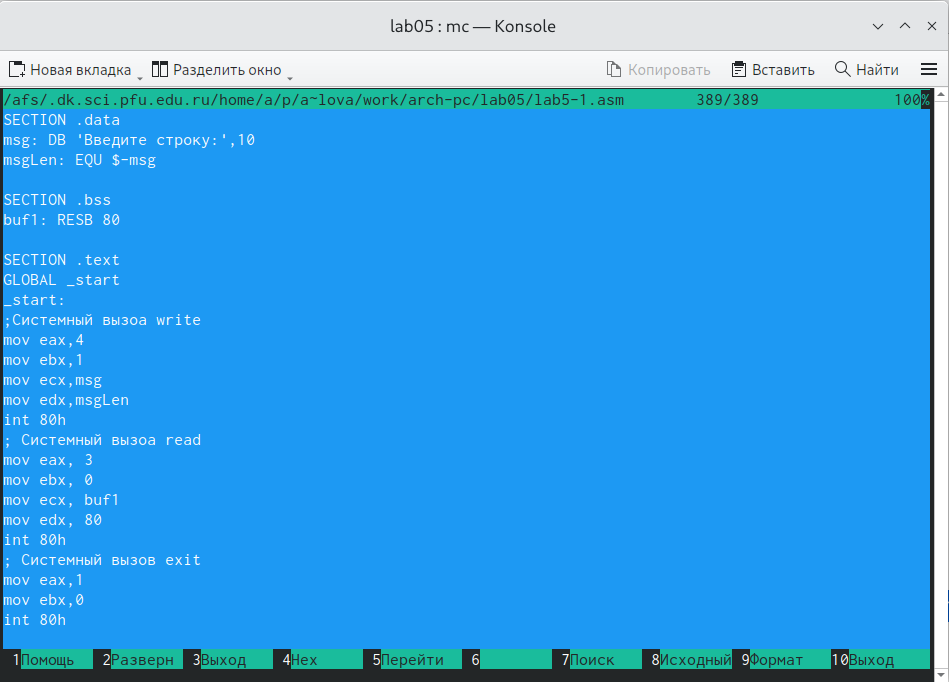
Введём текст программы вывода сообщения “Введите строку” на экран и ввода строки с клавиатуры (рис. ??).



Текст программы вывода сообщения на экран и ввода строки

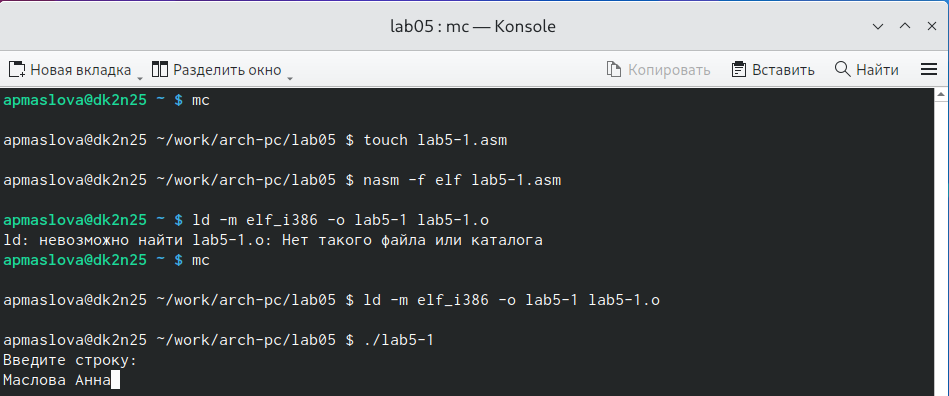
Сохранили изменения в файле с помощью функциональной клавиши F2 и закрыли mcedit, используя клавишу F10.

Нажмём клавишу F3, выделив файл *lab5-1.asm*, и проверим, что в нём содержится текст написанной программы (рис. ??).



Содержимое файла lab5-1.asm

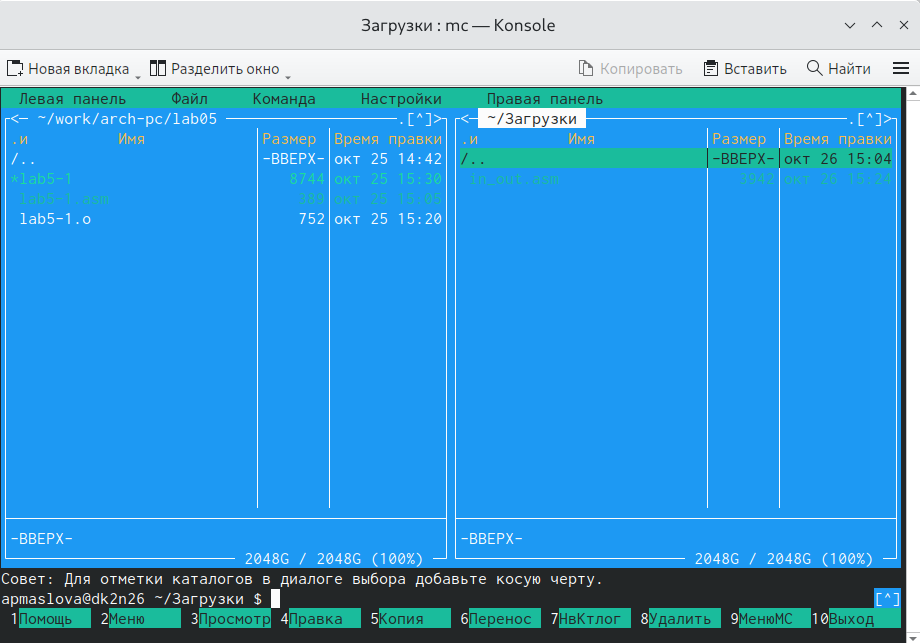
Теперь оттранслируем текст программы в объектный файл, а затем объектный файл скомпонуем и запустим получившийся исполняемый файл (рис. ??).



Транслирование, компоновка lab5-1.asm и запуск программы

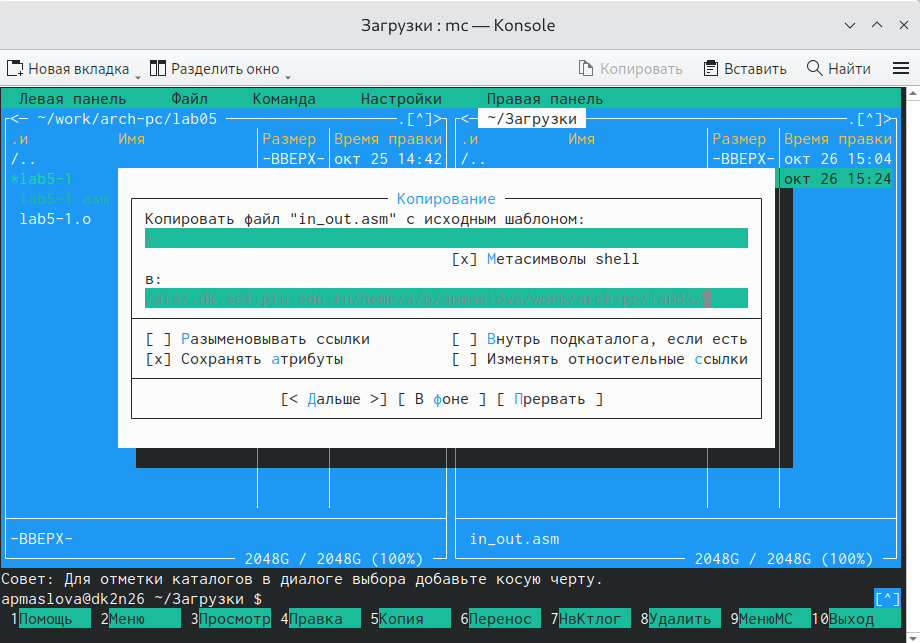
Как мы видим, программы работает: на экран выводится сообщение “Введите строку:” и считывается введённая строка (в нашем случае имя и фамилия - Маслова Анна).

Теперь создадим программу, выполняющую такую же функцию, но с помощью внешнего файла. Для этого с ТУИС скачиваем файл *in\_out.asm* (рис. ??).



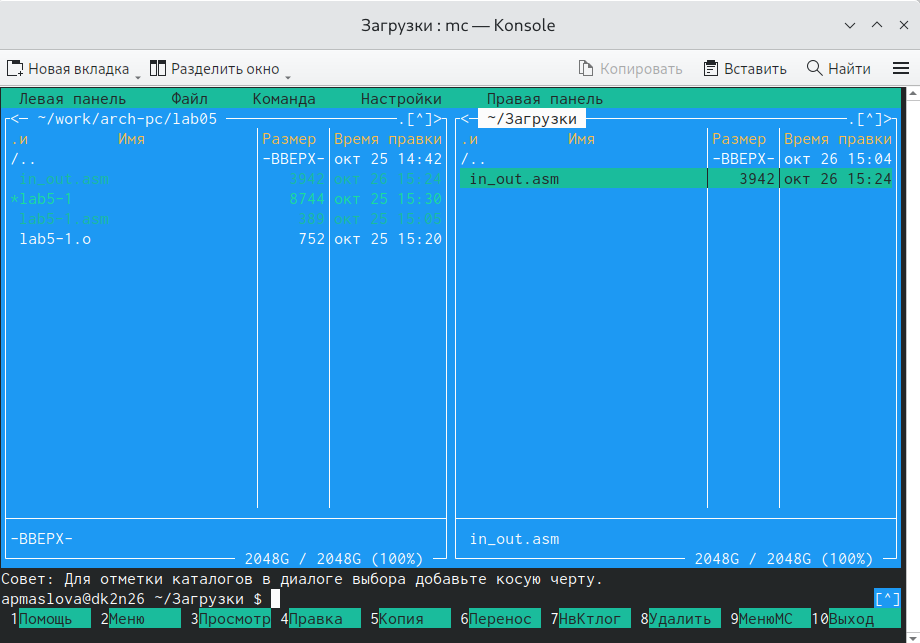
Загруженный файл in\_out.asm

Загруженный файл *in\_out.asm* помещён в каталог *Загрузки*. Чтобы применять его в программе *lab5-1.asm* он должен лежать в том же каталоге *~/work/arch-pc/lab05*. Скопируем этот файл в нужный каталог с помощью функциональной клавиши F5 (рис. ??).



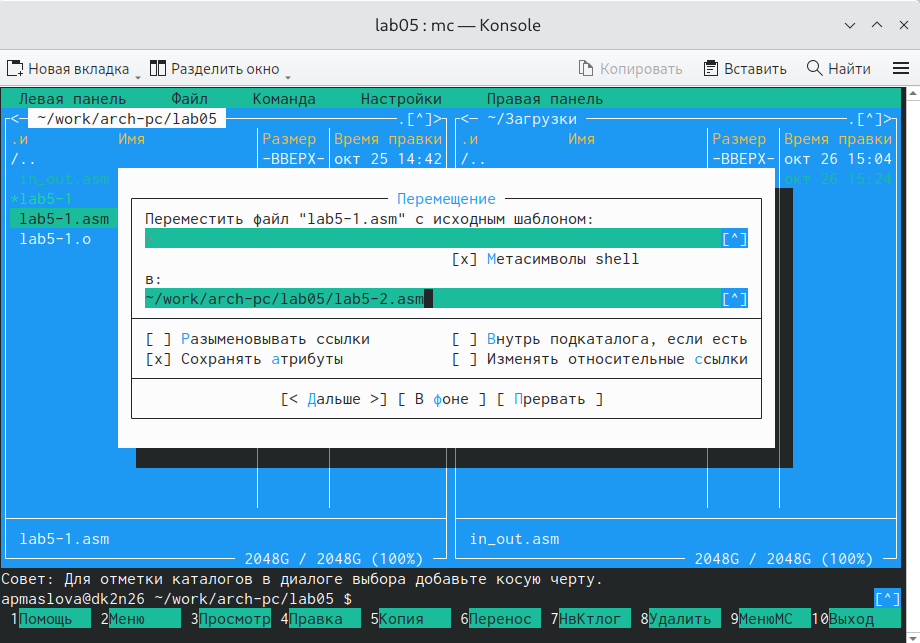
Копирование файла in\_out.asm в каталог lab05

На рис. ?? можем видеть, что файл *in\_out.asm* содержится в каталоге *~/work/arch-pc/lab05*.

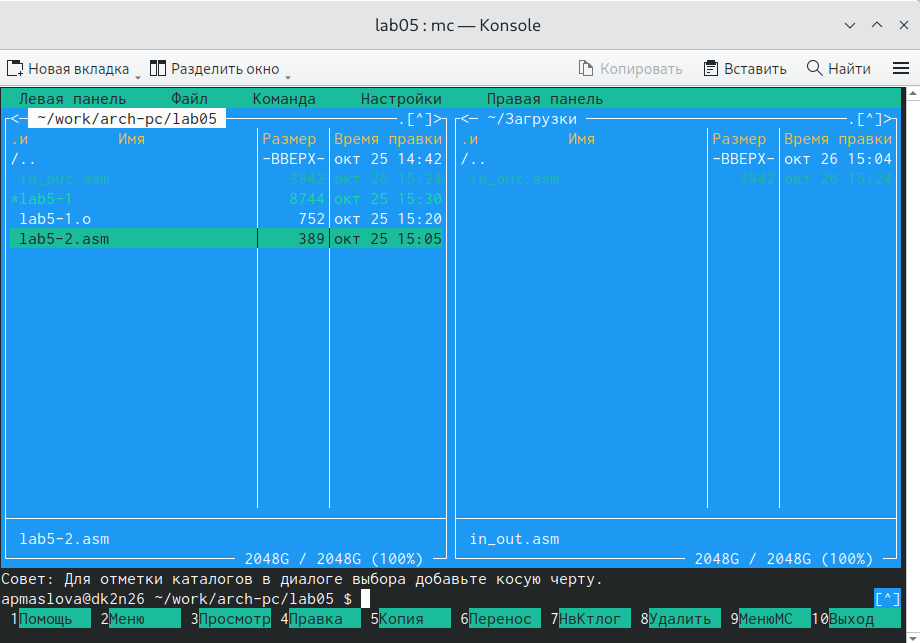


Файл in\_out.asm в каталоге lab05

Далее нам нужно создать файл *lab5-2.asm*. Выделим файл *lab5-1.asm*. С помощью клавиши F6 мы можем переместить (то есть заменить) файл *lab5-1.asm* на файл *lab5-2.asm* (рис. ?? , рис. ??).

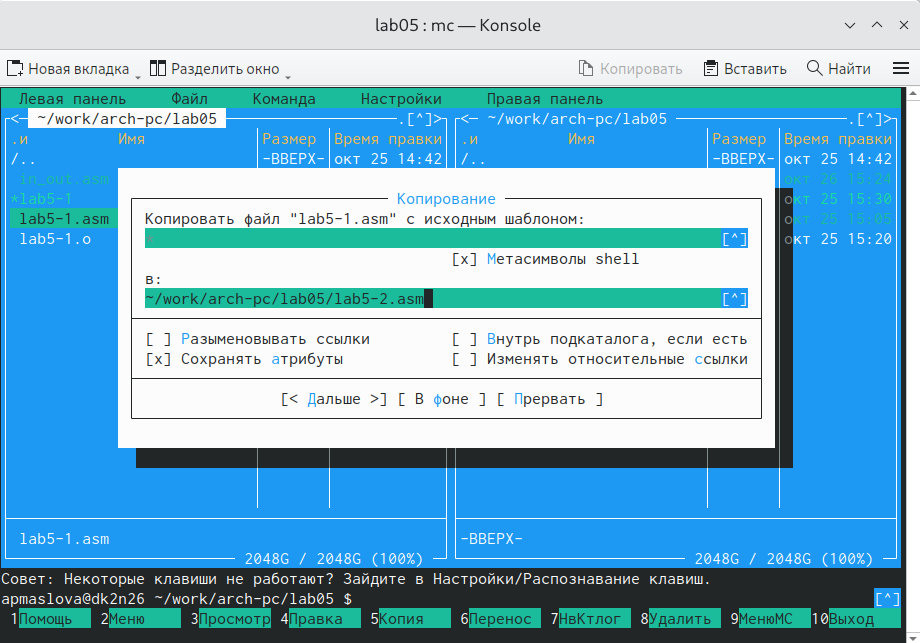


Перемещение файла lab5-1.asm

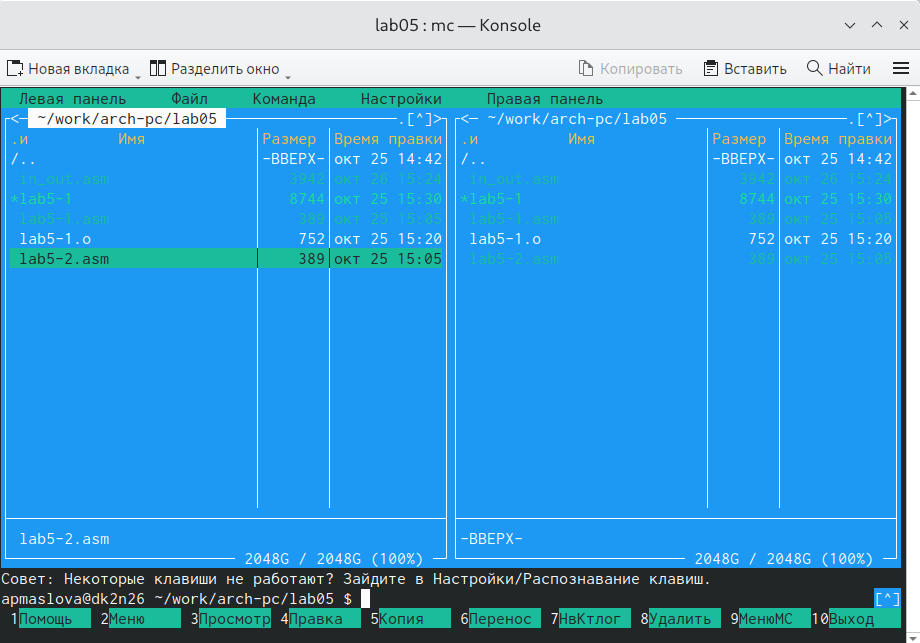


Файл lab5-1.asm заменён на lab5-2.asm

Однако файл *lab5-1.asm* нам пригодится в ходе выполнения дальнейших заданий лабораторной работы, поэтому вместо перемещения клавишей F6 файл *lab5-1.asm* мы скопируем в этот же каталог с помощью клавиши F5 и зададим ему имя “*lab5-2.asm*” (рис. ?? , рис. ??).

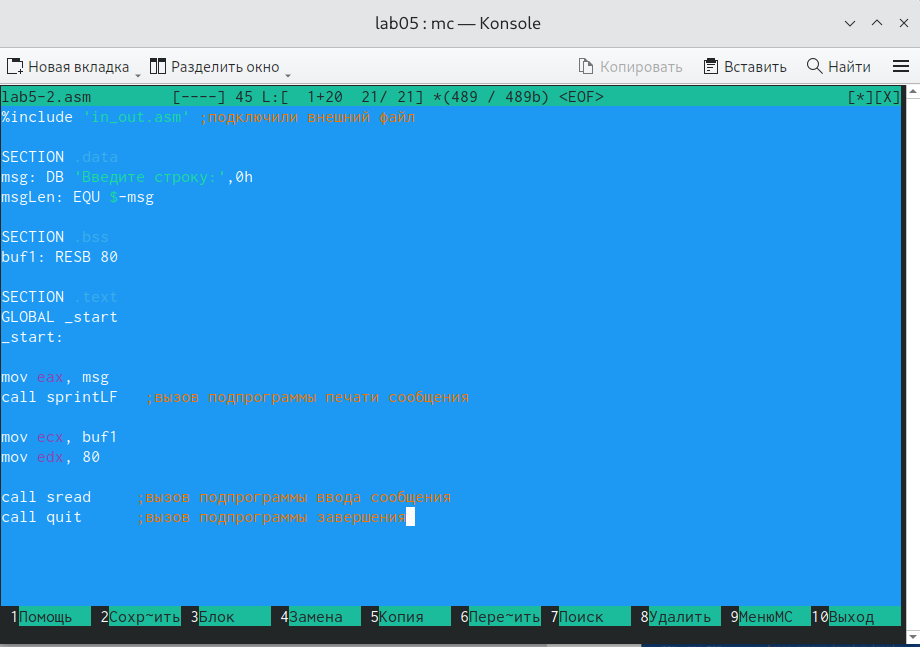


Копирование файла lab5-1.asm



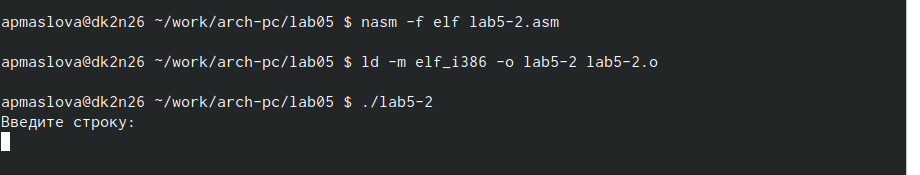
Копия файла lab5-1.asm с именем lab5-2.asm

Откроем созданный файл в редакторе и исправим текст программы в нём с использованием подпрограмм из загруженного внешнего файла *in\_out.asm*, а именно - подпрограмм *sprintLF, sread, guit* (рис. ??).



Текст программы в файле lab5-2.asm

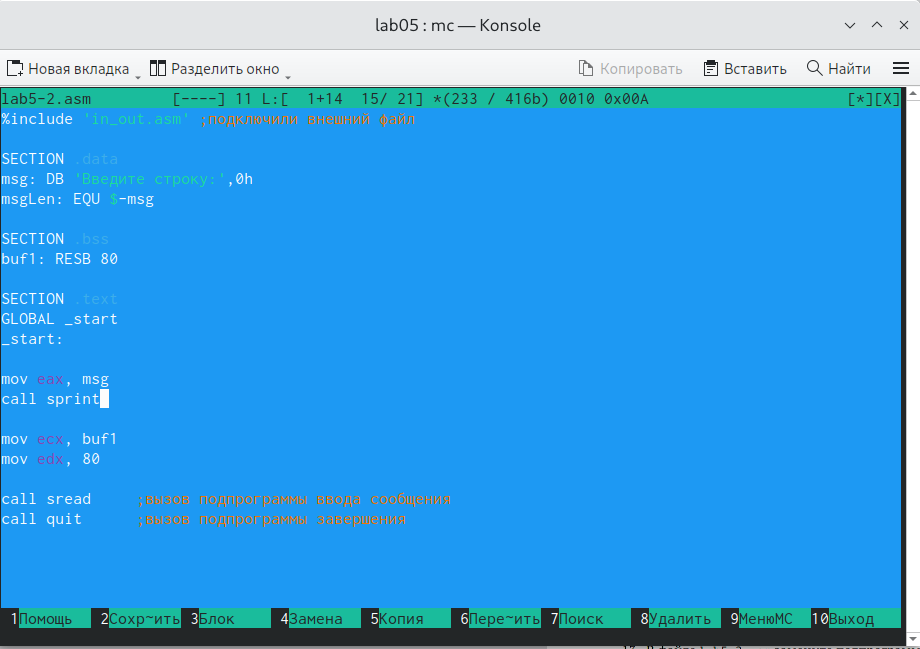
Создадим исполняемый файл и проверим его работу (рис. ??).



Запуск lab5-2

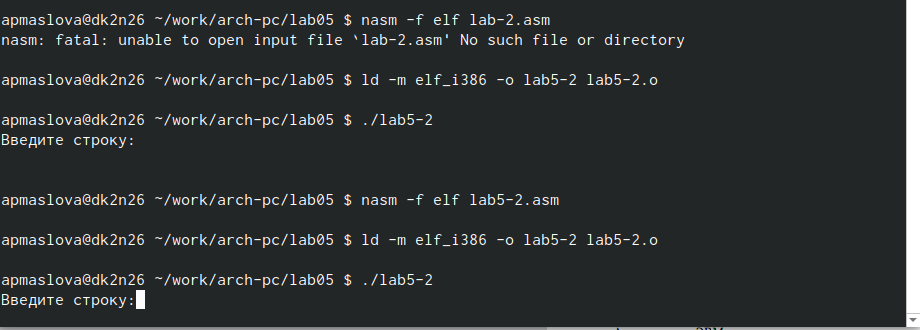
Как мы видим, программа выполняет свою функцию.

А теперь снова откроем в редакторе файл *lab5-2.asm* и заменим в нём подпрограмму *sprintLF* на *sprint* (рис. ??).



Замена sprintLF на sprint

Создадим исполняемый файл и запустим его (рис. ??).

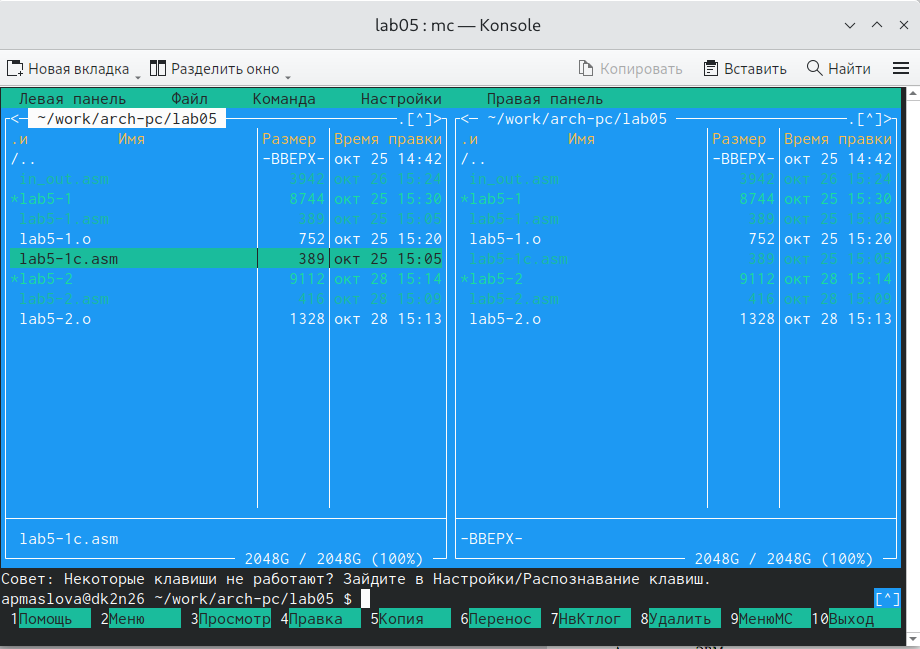


Запуск lab5-2 с исправленной подпрограммой

Файл запустился. Можно заметить, что подпрограмма *sprint* в отличие от подпрограммы *sprintLF* не переносит строку. То есть в первом случае мы вводили нашу строку “Маслова Анна” на следующей строке после сообщения “Введите строку:”, а во втором случае - в той же.

# 3 Выполнение заданий для самостоятельной работы

Создадим копию файла *lab5-1.asm* и зададим имя новому файлу “*lab5-1c.asm*”. Для этго используем клавишу F5 (рис. ??).



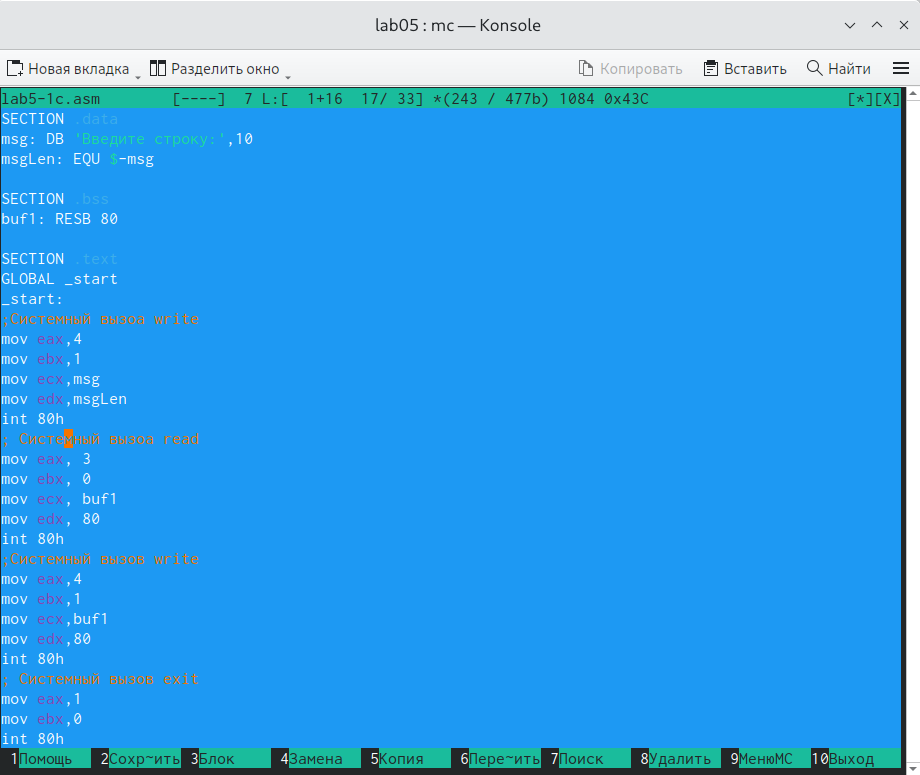
Созданный файл lab5-1c.asm

Файл создан. Откроем его в редакторе mcedit с помощью клавиши F4.

Изменим текст программы, не используя при этом внешний файл *in\_out.asm*. Нам нужно, чтобы теперь после ввода строки с клавиатуры, программа выводила введённую строку на экран. Для этого используем системный вызов *write* (в регистр *ecx* поместим значение переменной, куда записали введённую строку), как указано в листинге 3.1 и на рис. ?? .

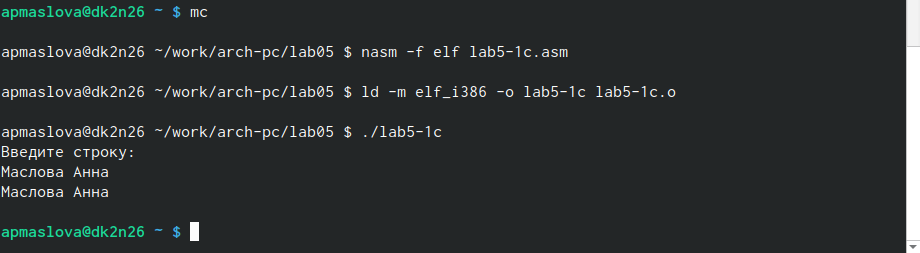
**Листинг 3.1. Программа вывода сообщения “Введите строку:” на экран, ввода строки с клавиатуры и вывода введённой строки на экран**

SECTION .data  
msg: DB 'Введите строку:',10  
msgLen: EQU $-msg  
  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
GLOBAL \_start  
\_start:  
  
;Системный вызоа write  
  
mov eax,4  
mov ebx,1  
mov ecx,msg  
mov edx,msgLen  
int 80h  
  
; Системный вызоа read  
  
mov eax, 3  
mov ebx, 0  
mov ecx, buf1  
mov edx, 80  
int 80h  
  
;Сиcтемный вызов write  
  
mov eax,4  
mov ebx,1  
mov ecx,buf1  
mov edx,80  
int 80h  
  
; Системный вызов exit  
  
mov eax,1  
mov ebx,0  
int 80h



Текст программы в файле lab5-1c.asm

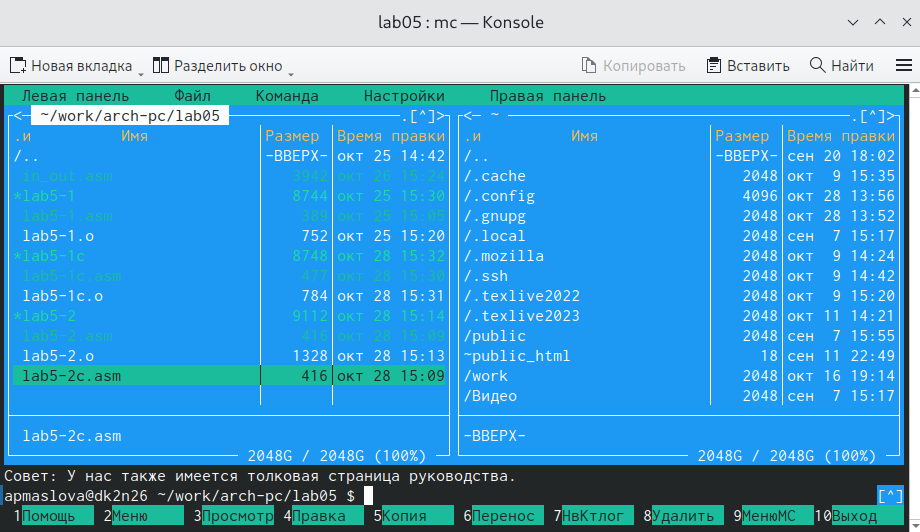
Создадим исполняемый файл и проверим его работу (рис. ??).



Запуск lab5-1c

Как мы видим, программа работает корректно: на экран выводится сообщение “Введите строку:”, мы вводим строку, и эта строка затем выводится на экран.

Теперь создадим копию файла *lab5-2.asm* и зададим имя новому файлу “*lab5-2c.asm*”. Для этго используем клавишу F5 (рис. ??).



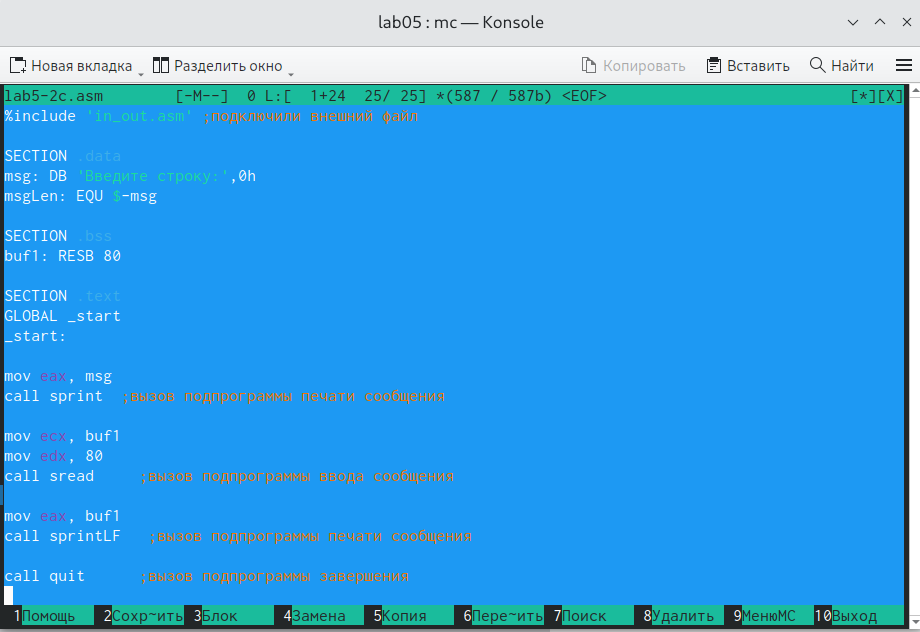
Созданный файл lab5-2c.asm

Файл создан. Откроем его в редакторе mcedit с помощью клавиши F4.

Изменим текст программы,используя внешний файл *in\_out.asm*. Нам нужно, чтобы программа работала так же, как *lab5-1c*. Для этого используем подпрограмму печати сообщения *sprintLF* (в регистр *ecx* поместим значение переменной, куда записали введённую строку), как указано в листинге 3.2 и на рис. ?? .

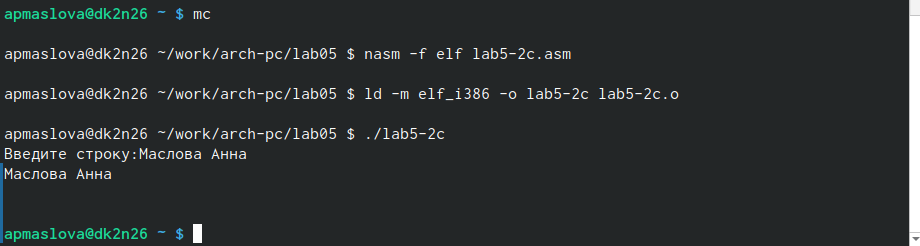
**Листинг 3.2. Программа вывода сообщения “Введите строку:” на экран, ввода строки с клавиатуры и вывода введённой строки на экран с использованием внешнего файла in\_out.asm**

%include 'in\_out.asm' ;подключили внешний файл  
  
SECTION .data  
msg: DB 'Введите строку:',0h  
msgLen: EQU $-msg  
  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
GLOBAL \_start  
\_start:  
  
mov eax, msg  
call sprint ;вызов подпрограммы печати сообщения  
  
mov ecx, buf1  
mov edx, 80  
call sread ;вызов подпрограммы ввода сообщения  
  
mov eax, buf1  
call sprintLF ;вызов подпрограммы печати сообщения  
  
call quit ;вызов подпрограммы завершения



Текст программы в файле lab5-2c.asm

Создадим исполняемый файл и проверим его работу (рис. ??).



Запуск lab5-2c

Как мы видим, программа работает корректно.

# 4 Выводы

Мы научились работать в *Midnight Commander* на практике. Освоили инструкции языка ассемблера *mov* и *int*.

# Список литературы

1. GDB: The GNU Project Debugger. — URL: https://www.gnu.org/software/gdb/.
2. GNU Bash Manual. — 2016. — URL: https://www.gnu.org/software/bash/manual/.
3. Midnight Commander Development Center. — 2021. — URL: https://midnight-commander. org/.
4. NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: http://www.amazon.com/Learning- bash-Shell-Programming-Nutshell/dp/0596009658.
6. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс,