# Лабораторная работа номер 5.

**Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM. Системные вызовы в ОС GNU Linux**

Сорокин Кирилл

# Содержание

# Список иллюстраций

# 1 Цель работы

Начиться работать с Midnight Commander, а также освоить инструкции mov и int в языке ассемблера.

# 2 Задание

Используя МС, написать программы, основываясь на приведённых материалах, а затем выполнить самостоятельную работу.

# 3 Теоретическое введение

Midnight Commander (или просто mc) — это программа, которая позволяет просматривать структуру каталогов и выполнять основные операции по управлению файловой системой, т.е. mc является файловым менеджером. Midnight Commander позволяет сделать работу с файлами более удобной и наглядной. Инструкция языка ассемблера mov предназначена для дублирования данных источника в приёмнике. Инструкция языка ассемблера int предназначена для вызова прерывания с указанным номером.

# 4 Выполнение лабораторной работы

Откроем Midnight Commander командой mc (рис. 4.1).


Рис. 4.1: Открытие Midnight Commander

Перейдём в необходимую нам дерикторию(рис. 4.2).

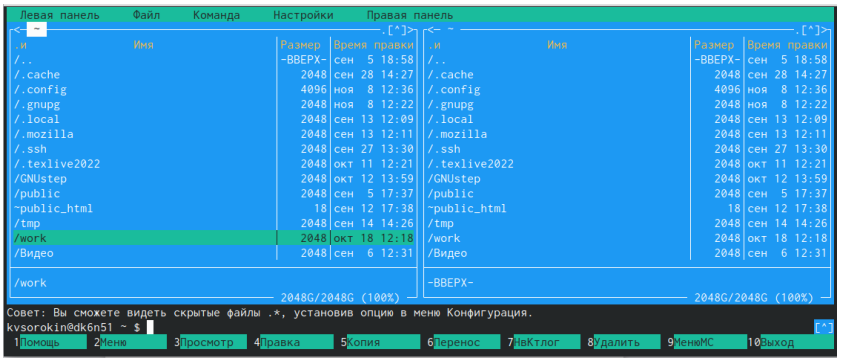


Рис. 4.2: Отображение папок в Midnight Commander

С помощью клавиши F7 создадим папку(рис. 4.3).

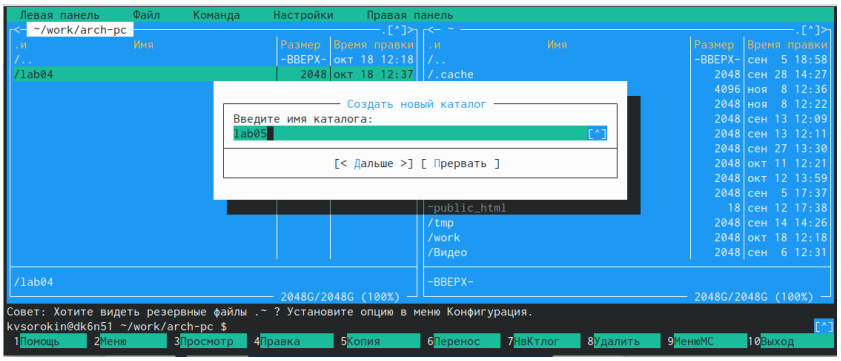


Рис. 4.3: Создание папки

Удостоверимся, что папка создана (рис. 4.4).

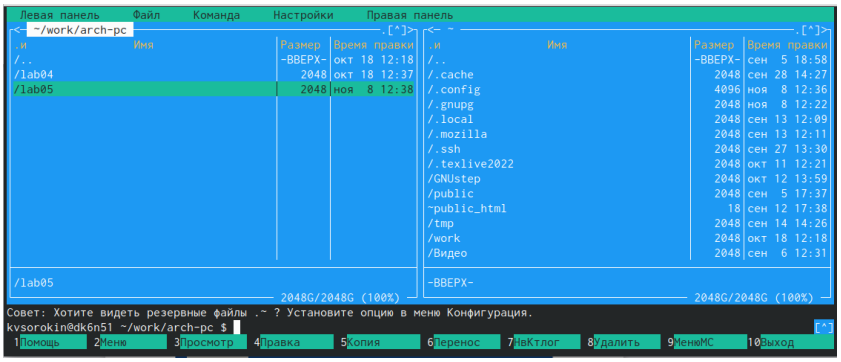


Рис. 4.4: Вид созданной папки

Командой touch создадим файл lab5-1.asm (рис. 4.5).


Рис. 4.5: Создание файла lab5-1.asm

Откроем файл для редактирование клавишой F4 (рис. 4.6).

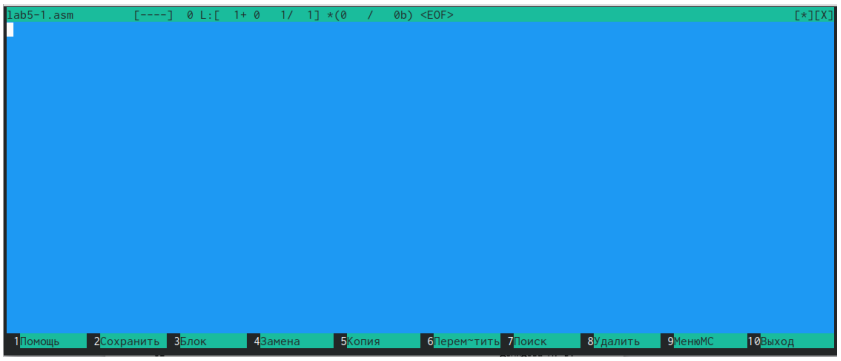


Рис. 4.6: Вид открытого файла

Введём текст программы в файл (рис. 4.7).

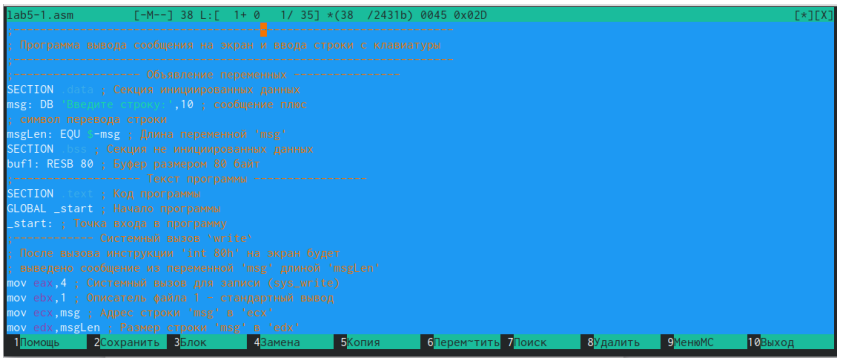


Рис. 4.7: Файл с текстом программы

Так как у нас редактор mcedit нажмем клавишу F2 для сохранения и клавишой F10 для выхода из редактирования файла (рис. 4.8).

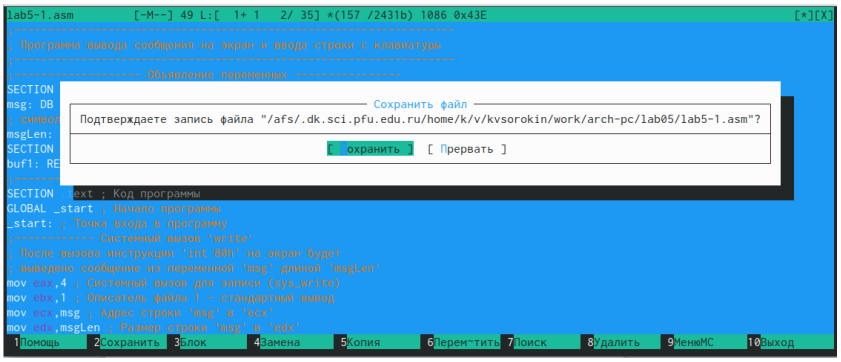


Рис. 4.8: Сохранение файла

Нажмём клавишу F3 для просмотра содежимого файла lab5-1.asm (рис. 4.9).

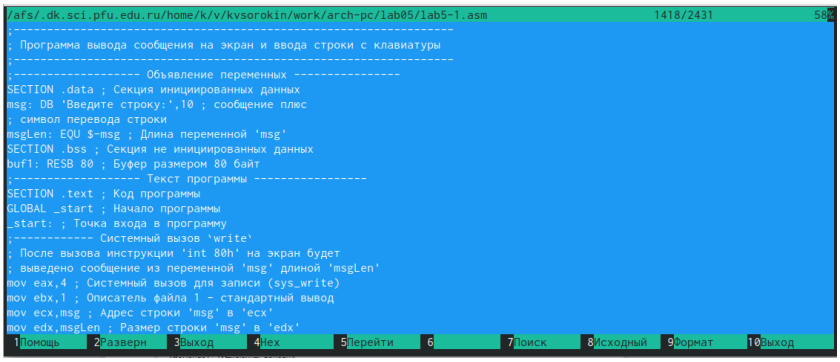


Рис. 4.9: Просмотр содержимого файла

Оттрансилруем текст lab5-1.asm в объектный файл, затем выполним его компоновку и запустим файл командой ./lab5-1 (рис. 4.10).

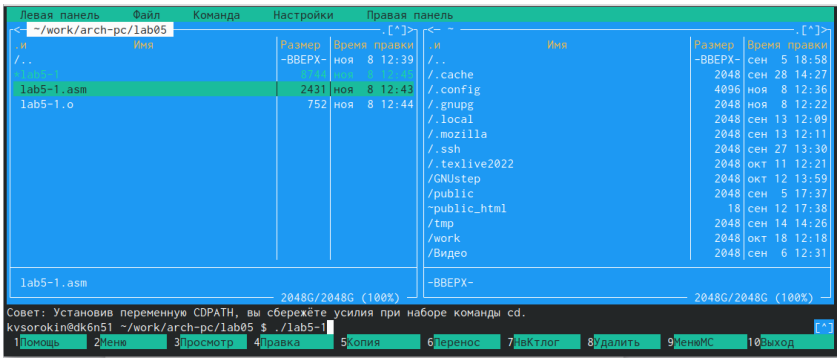


Рис. 4.10: Содержимое папки после создания всех файлов

Убедимся в правильности работы программы (рис. 4.11).

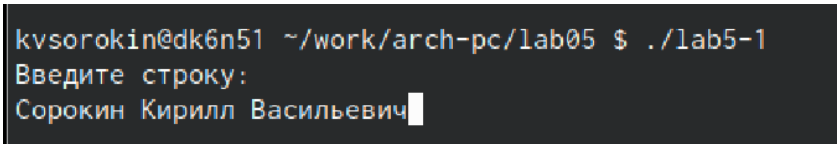


Рис. 4.11: Работа первой программы

Для удобства откроем дополнительную панель терминала и найдём заранее скаченный файл in_out.asm (рис. 4.12).

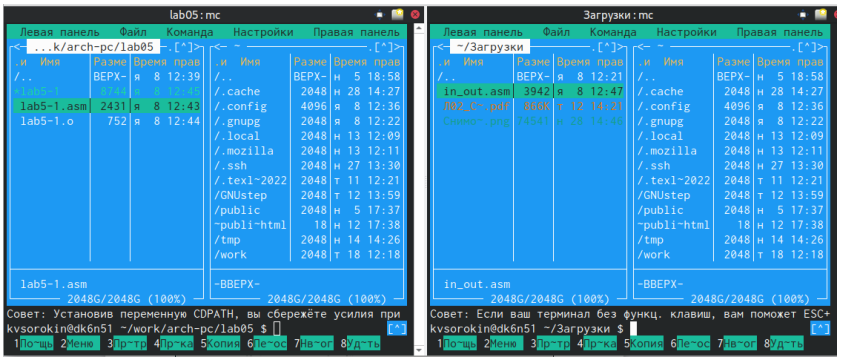


Рис. 4.12: Вид двойного терминала

Скопируем файл в наш рабочий каталог с помощью клавиши F5 (рис. 4.13).

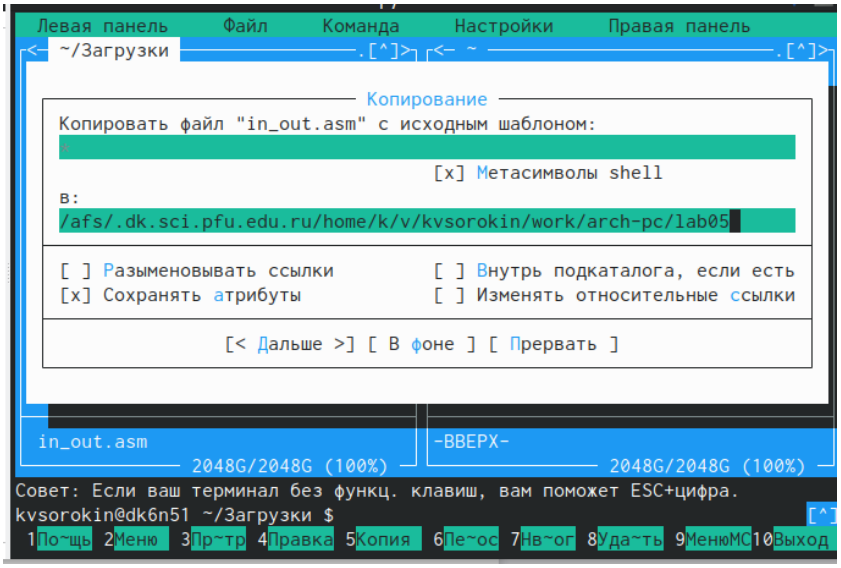


Рис. 4.13: Копирование файла in_out.asm

Убедимся, что он корректно перенёсся (рис. 4.14).

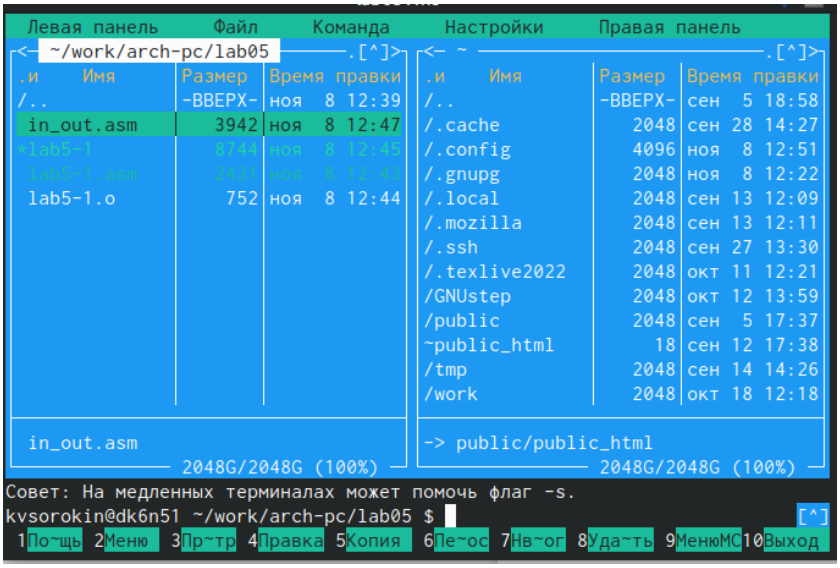


Рис. 4.14: Вид папки с in_out.asm

Создадим копию файла lab5-1.asm с именем lab5-2.asm (рис. 4.15).

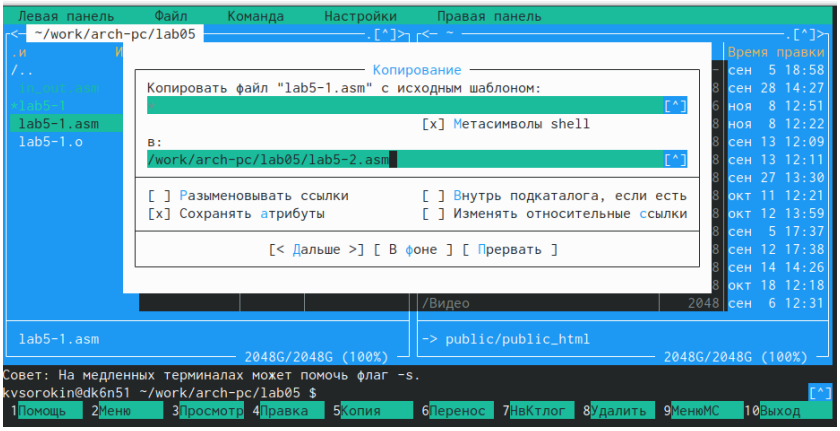


Рис. 4.15: Создание файла lab5-2.asm

Отредактируем его, так чтобы он использовал файл in_out.asm (рис. 4.16).


Рис. 4.16: Редактирование lab5-2.asm

Создадим его объектный файл и файл программы (рис. 4.17).

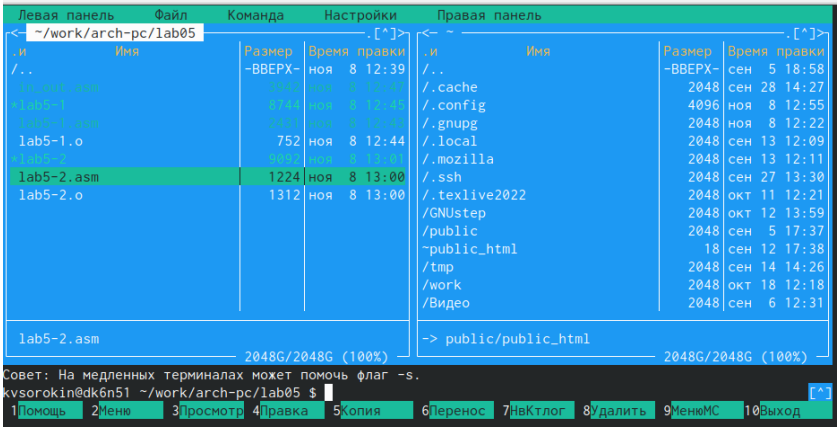


Рис. 4.17: Вид папки с новыми файлами

Проверим его работу (рис. 4.18).

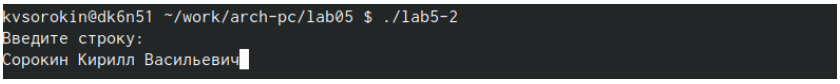


Рис. 4.18: Работа файла lab5-2.asm

Заменим подпрограмму sprintLF на sprint (рис. 4.19).

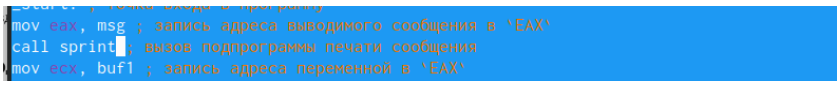


Рис. 4.19: Редактирование файла lab5-2.asm

Выполним новую программу и заметим, что теперь ввод текста идёт не с новой строки, а стой же на какой был напичатан изначальный текст (рис. 4.20).

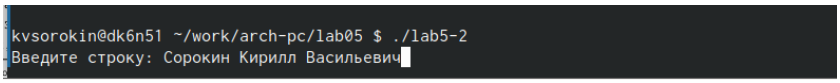


Рис. 4.20: Работа второй версии lab5-2

## 4.1 Самостоятельная работа

Создадим копию файла lab5-1.asm с именем lab5-1s.asm для самостоятельной работы (рис. 4.21).


Рис. 4.21: Начало самотоятельной работы

Для того чтобы новая программа выполняла дополнительные действия добавим вывод, ранее введённых данных, и запросим новый ввод, чтобы была возможность отследить результат вывода (рис. 4.22).

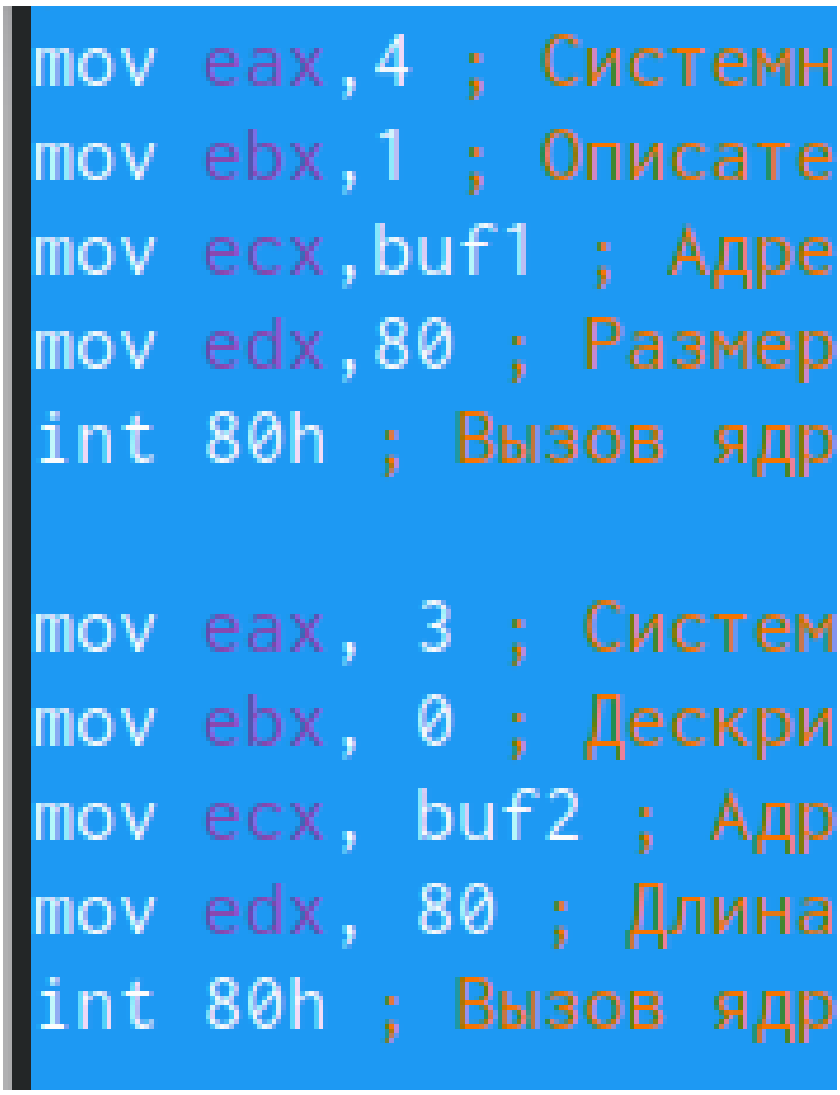


Рис. 4.22: Изменение первого файла

Создадим все файлы для работы программы (рис. 4.23).


Рис. 4.23: Файлы lab5-1s

Запустим программу и убедимся, что она корректно выполняет заданное задание (рис. 4.24).

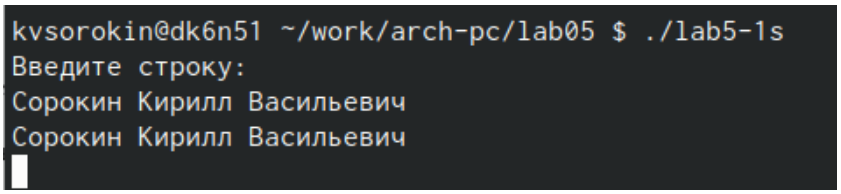


Рис. 4.24: Работа lab5-1s

Создадим копию файла lab5-2.asm с именем lab5-2s.asm для самостоятельной работы (рис. 4.25).

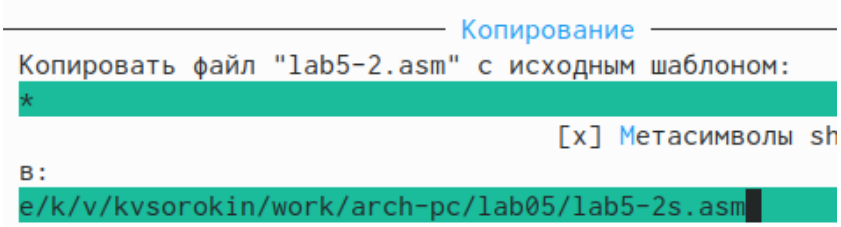


Рис. 4.25: Копирование второго файла

От редактируем новую программу. Чтобы она могла выполнить все необходимые действия добавим вывод, ранее введённых данных, и запросим новый ввод, чтобы была возможность отследить результат вывода (рис. 4.26).

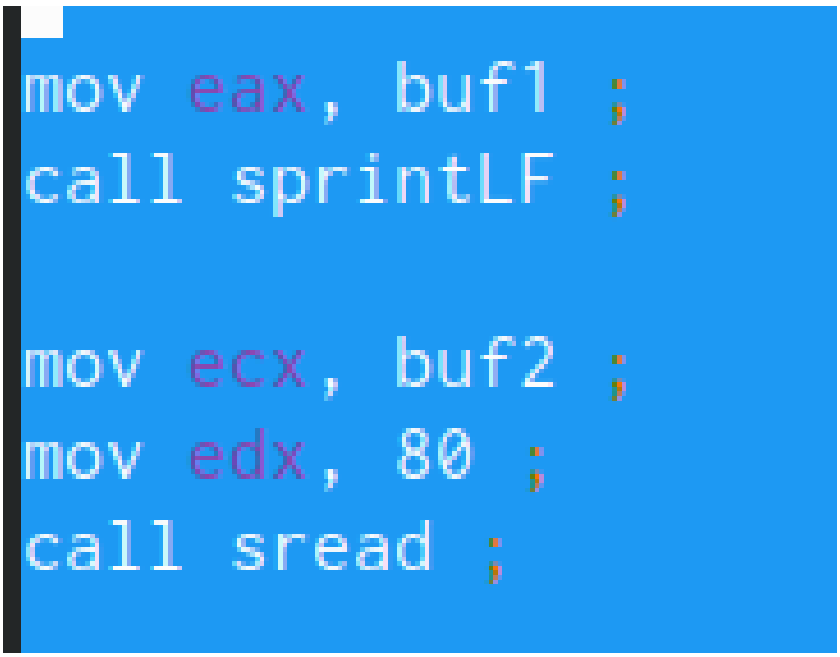


Рис. 4.26: Изменение второго файла

Создадим все файлы для работы программы (рис. 4.27).

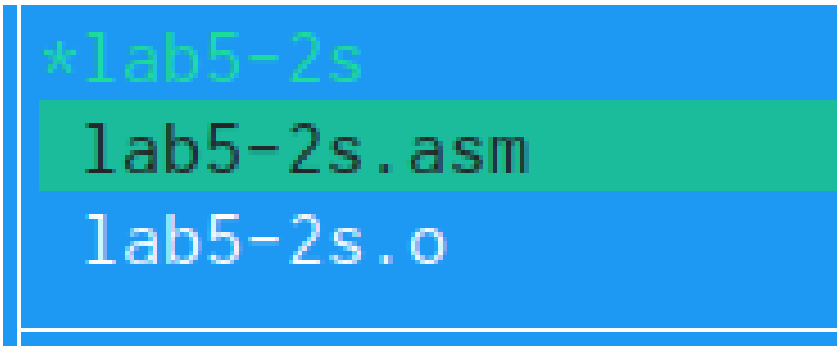


Рис. 4.27: Файлы lab5-2s

Запустим программу и убедимся, что она корректно выполняет заданное задание, но из-за того, что в изначальном файле lab5-2.asm мы оставили sprint, а не sprintLF, то вид ввода немного отличается от lab5-1s (рис. 4.28).

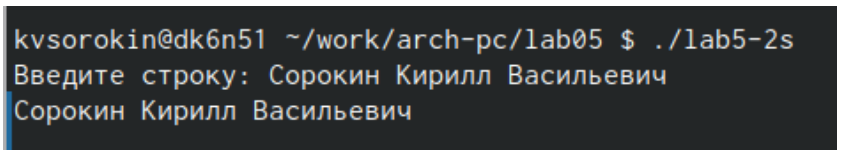


Рис. 4.28: Работа lab5-2s

# 5 Выводы

Мы научились использовать Midnight Commander, а также написали несколько программ на языке ассемблера с использованием новых инструкций.

# 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://midnightcommander.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, 2017.
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: https://www.opennet.ru/docs/RUS/nasm/.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm_unix.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. -

874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. -СПб. : Питер,
17. — 1120 с. — (Классика Computer Science)