Отчёт по лабораторной работе №8

Дисциплина: Архитектура компьютера

Бызова Мария Олеговна

Содержание

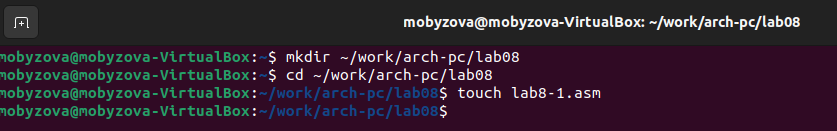
# 1 Цель работы

Целью данной лабораторной работы является приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

# 2 Выполнение лабораторной работы

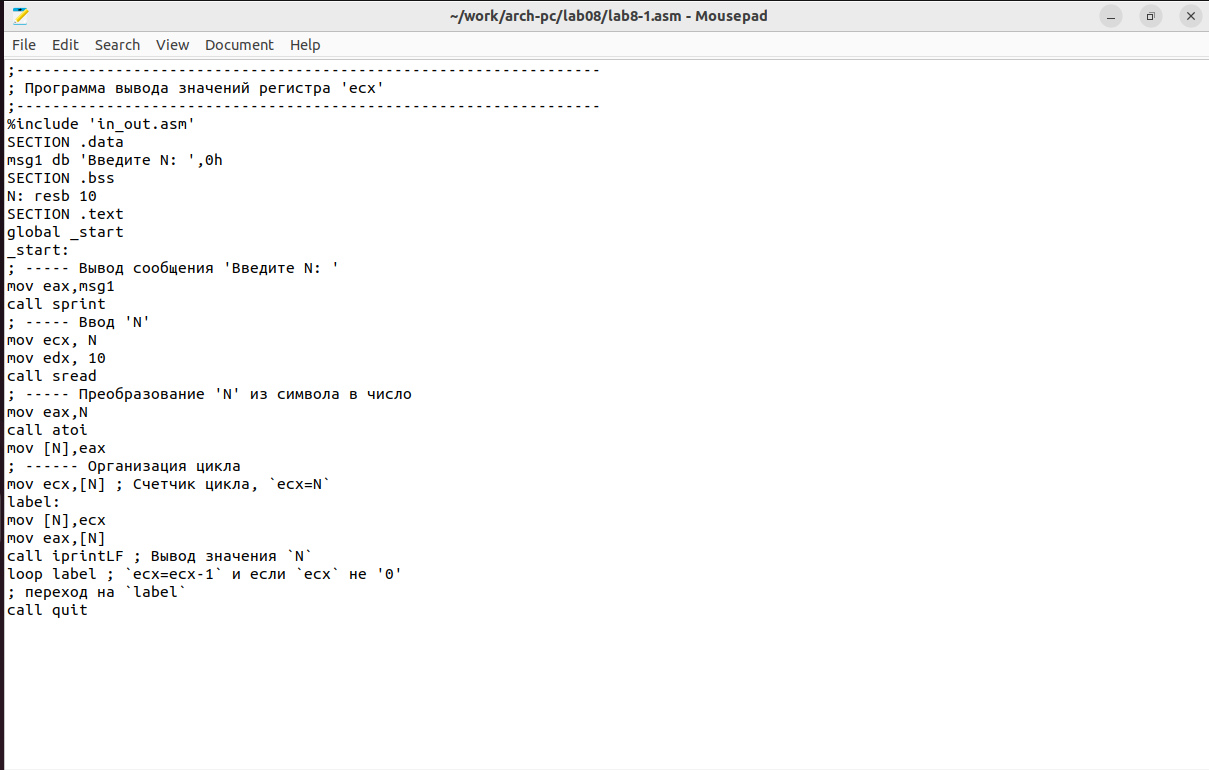
## 2.1 Реализация циклов в NASM

С помощью утилиты mkdir создаем директорию, в которой будем создавать файлы с программами для лабораторной работы №8. Переходим в созданный каталог с помощью утилиты cd. С помощью утилиты touch создаем файл lab8-1.asm (рис. [??]).



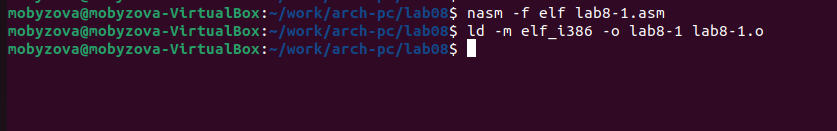
Создание необходимой директории и файла

При помощи Midnight Commander открываем созданный файл lab8-1.asm, вставляем в него программу вывода значений регистра ecx из листинга 8.1 (рис. [??]).

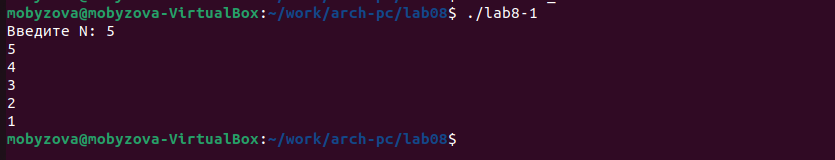


Редактирование файла

Далее создаем исполняемый файл и запускаем его (рис. [??], [??]).

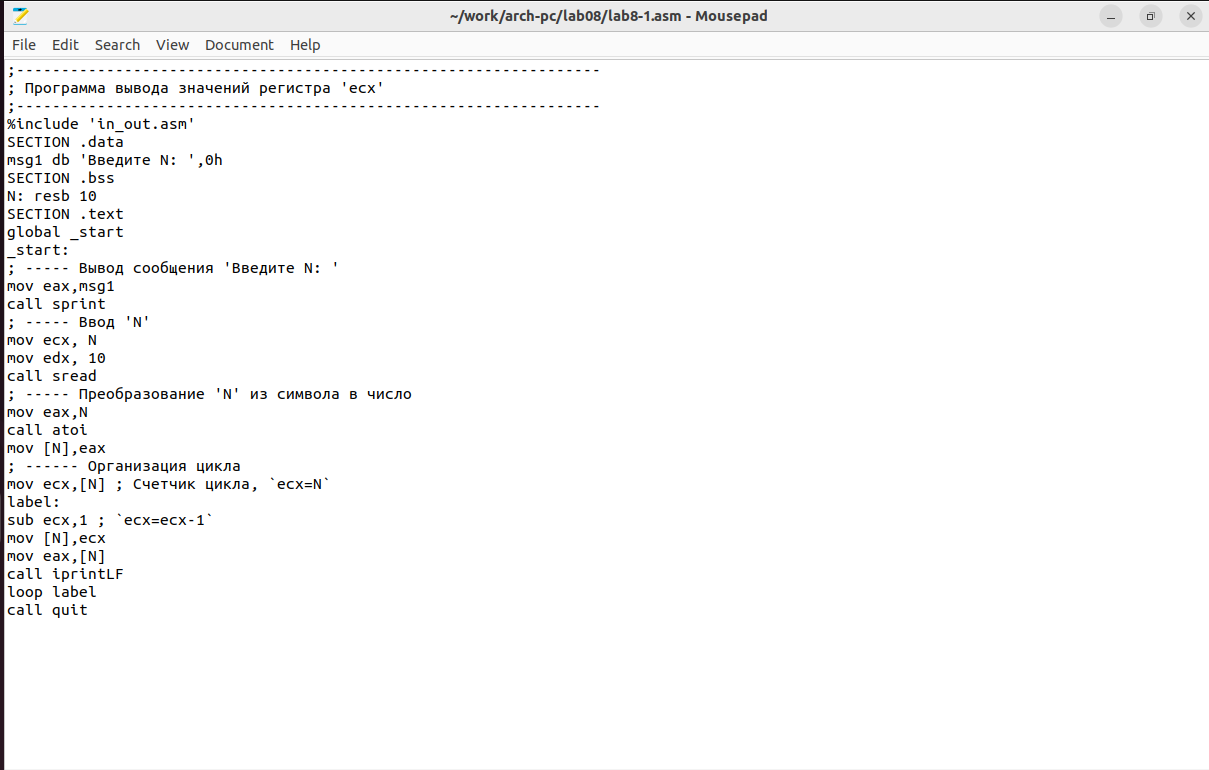


Создание исполняемого файла



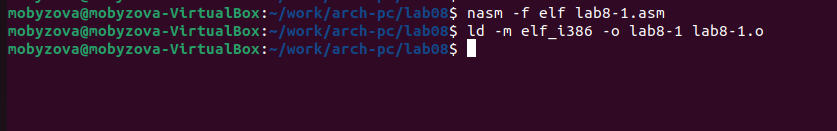
Запуск исполняемого файла

Далее мы изменяем текст программы, добавив изменение значение регистра ecx в цикле (рис. [??]).

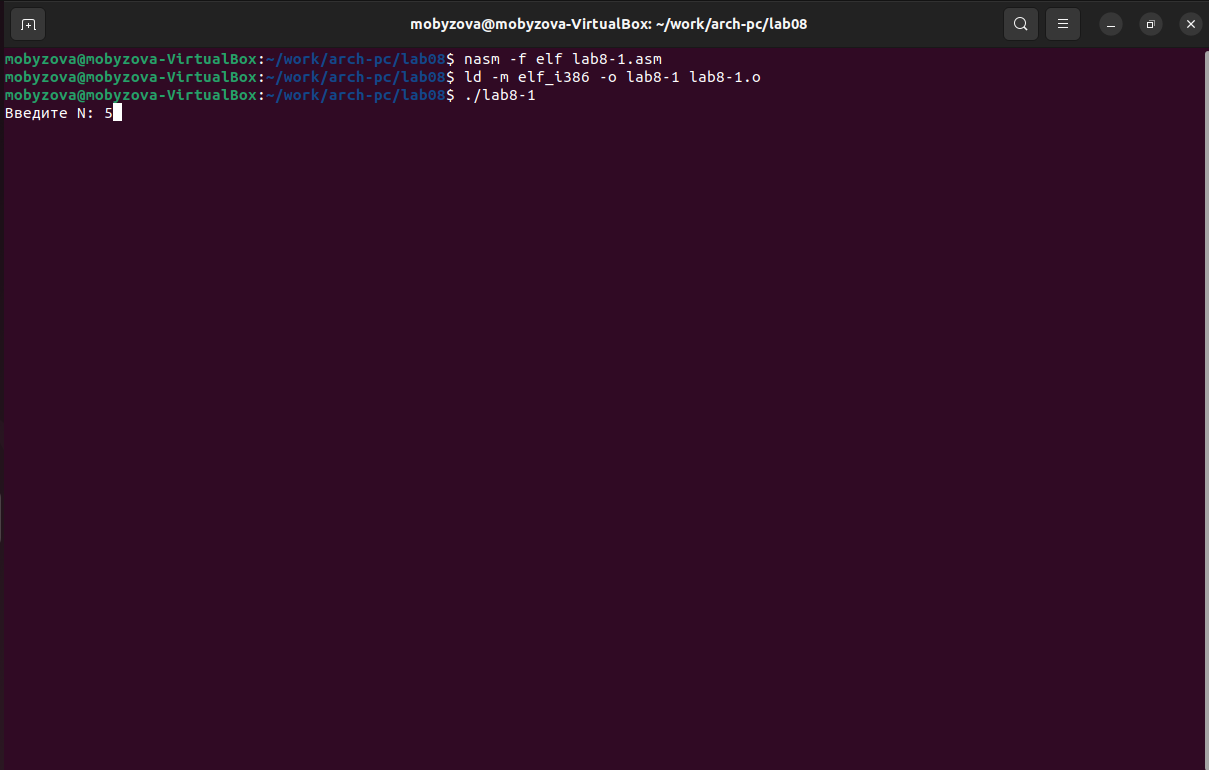


Редактирование файла

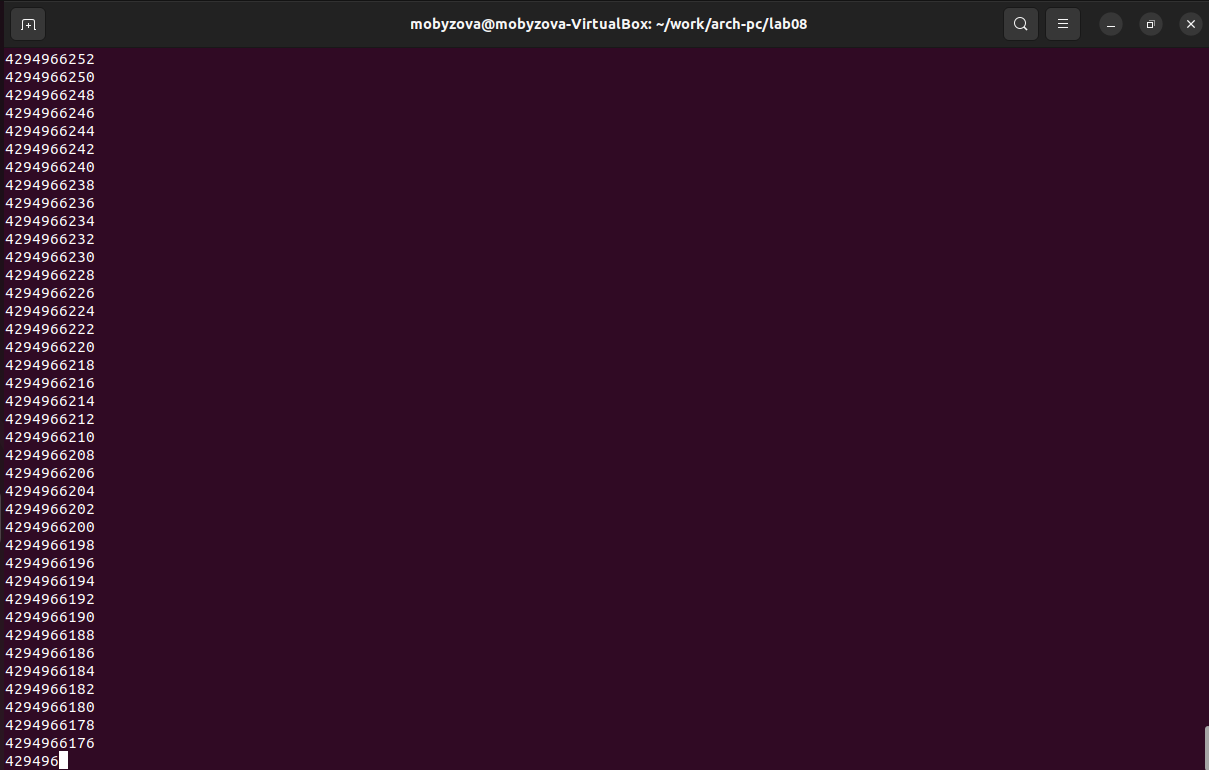
Далее создаем исполняемый файл и запускаем его (рис. [??], [??], [??]).



Создание исполняемого файла



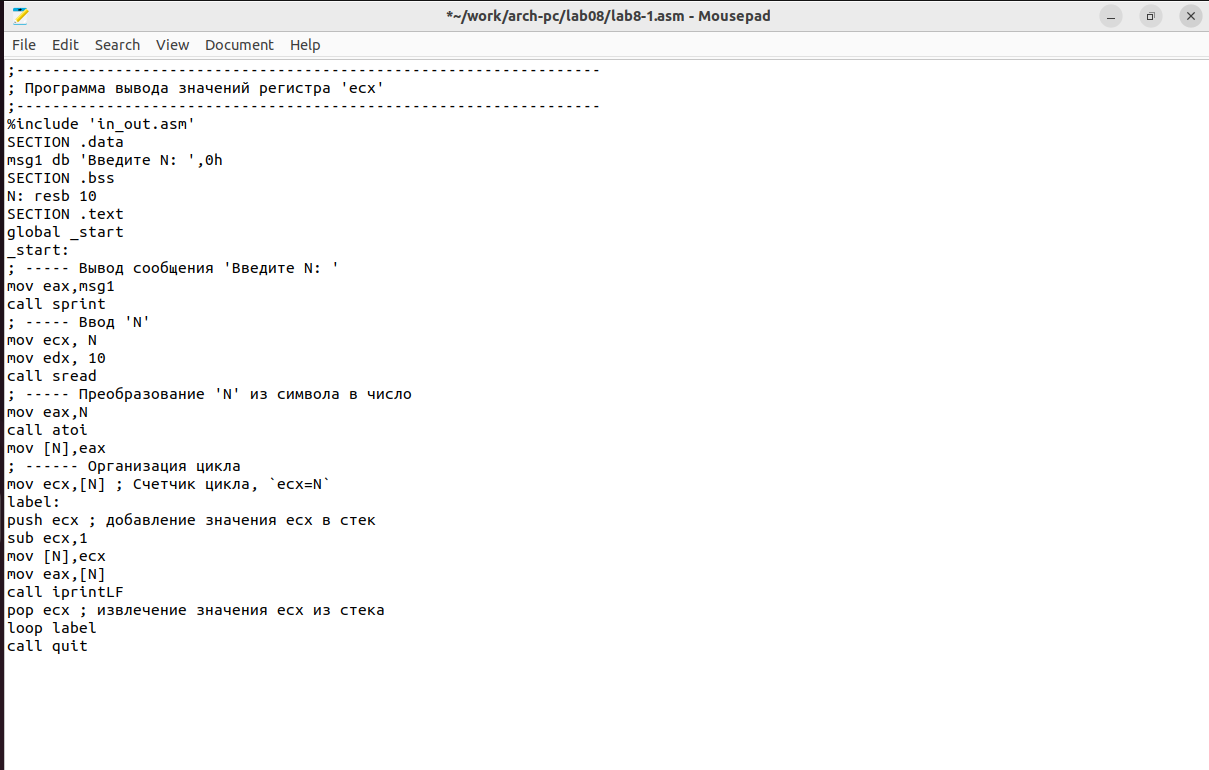
Запуск исполняемого файла



Запуск исполняемого файла

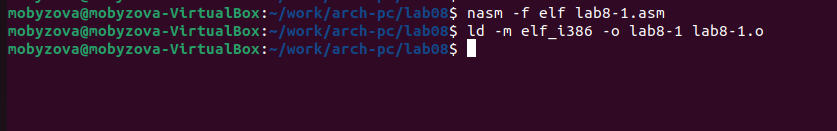
Таким образом, использование регистра ecx в теле цилка loop может привести к некорректной работе программы. Регистр ecx принимает в цикле некорректные значения, а число проходов цикла не соответсвует значению N введенному с клавиатуры.

Затем мы внесём изменения в текст программы добавив команды push и pop (добавления в стек и извлечения из стека) для сохранения значения счетчика цикла loop (рис. [??]).

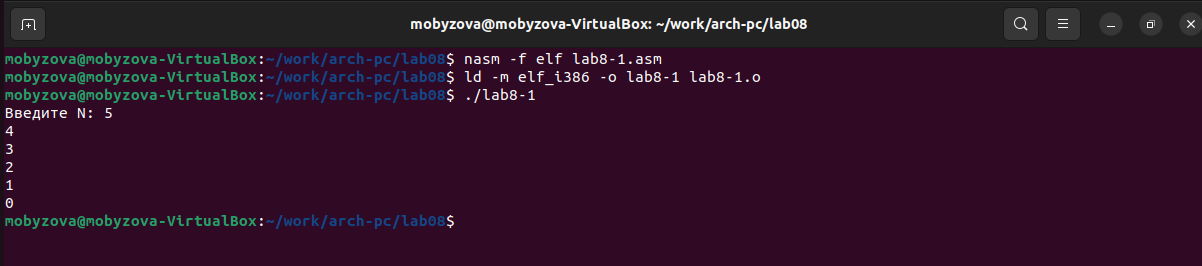


Редактирование файла

Далее создаем исполняемый файл и запускаем его (рис. [??], [??])



Создание исполняемого файла



Запуск исполняемого файла

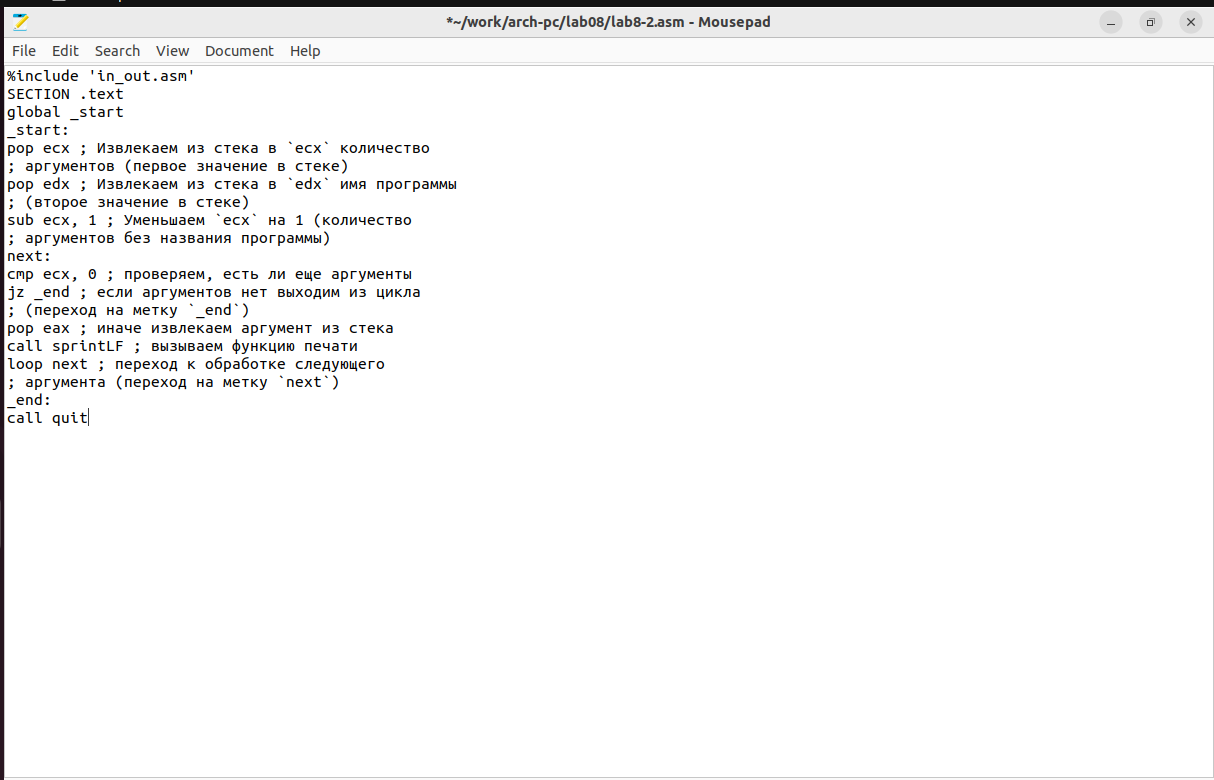
В данном случае число проходов цикла соответсвуют значению N введенному с клавиатуры, однако выводимые на экран значения будут принимать вид от “4” до “0” соответственно.

Создаем файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 (рис. [??]).

Создание файла

Создание файла

Внимательно изучив текст программы вычисления суммы аргументов командной строки из листинга 8.2, введем его в lab8-2.asm (рис. [??]).

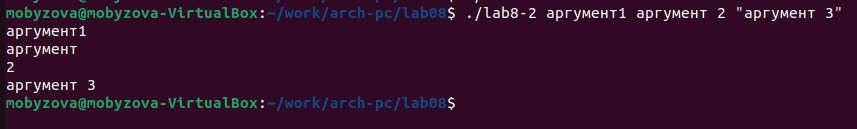


Редактирование файла

Далее создаем исполняемый файл и запускаем его (рис. [??], [??])

Создание исполняемого файла

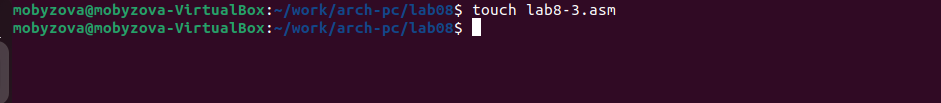
Создание исполняемого файла



Запуск исполняемого файла

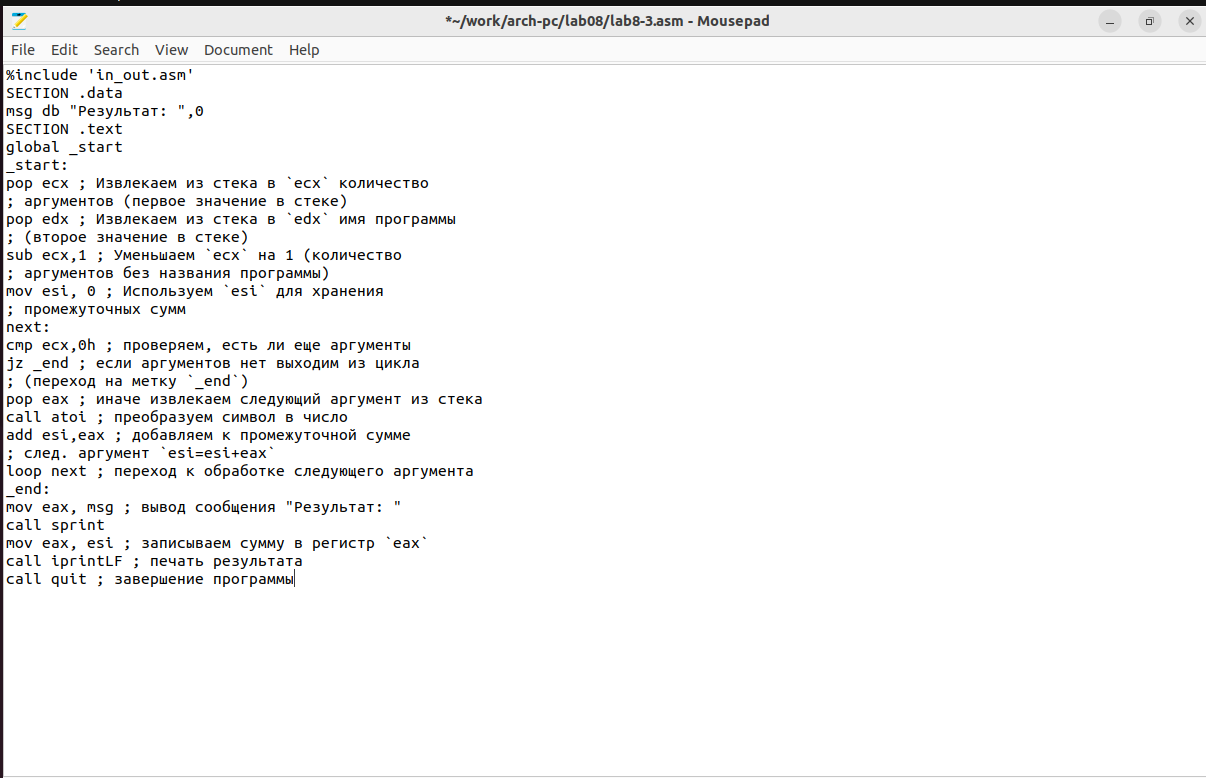
Программа обрабатывает и выводит на экран 4 аргумента.

Создаем файл lab8-3.asm в каталоге ~/work/arch-pc/lab08 (рис. [??]).



Создание файла

Внимательно изучив текст программы вычисления суммы аргументов командной строки из листинга 8.3, введем его в lab8-3.asm (рис. [??]).

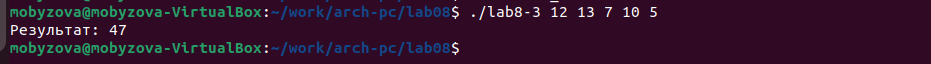


Редактирование файла

Далее создаем исполняемый файл и запускаем его (рис. [??], [??])

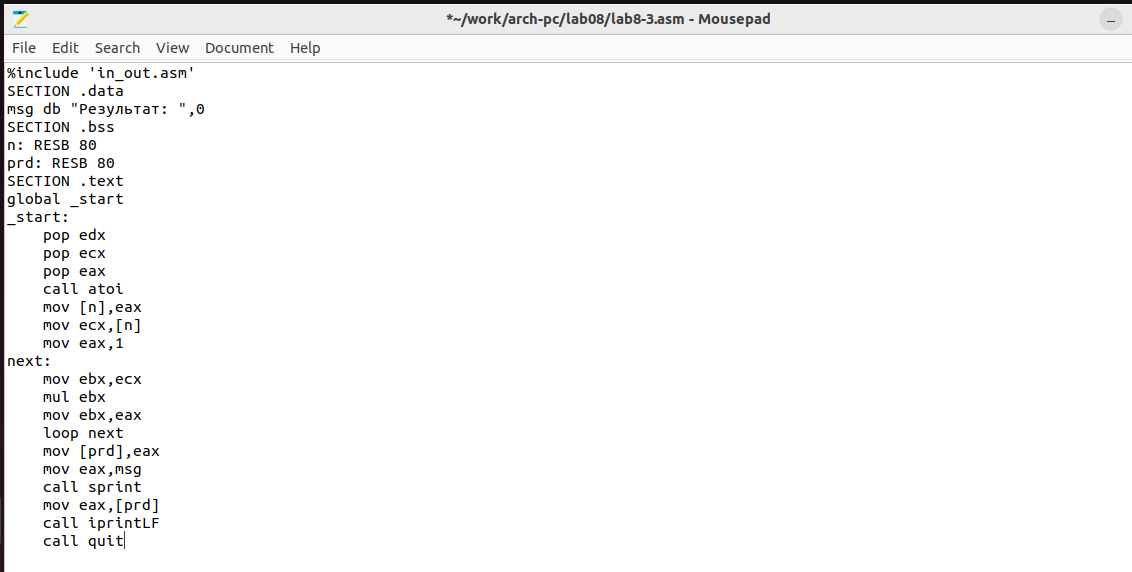
Создание исполняемого файла

Создание исполняемого файла



Запуск исполняемого файла

Изменим текст программы из листинга 8.3 для вычисления произведения аргументов командной строки (рис. [??]).



Редактирование файла

Далее создаем исполняемый файл и запускаем его (рис. [??], [??])

Создание исполняемого файла

Создание исполняемого файла

Запуск исполняемого файла

Запуск исполняемого файла

# 3 Выполнение заданий для самостоятельной работы

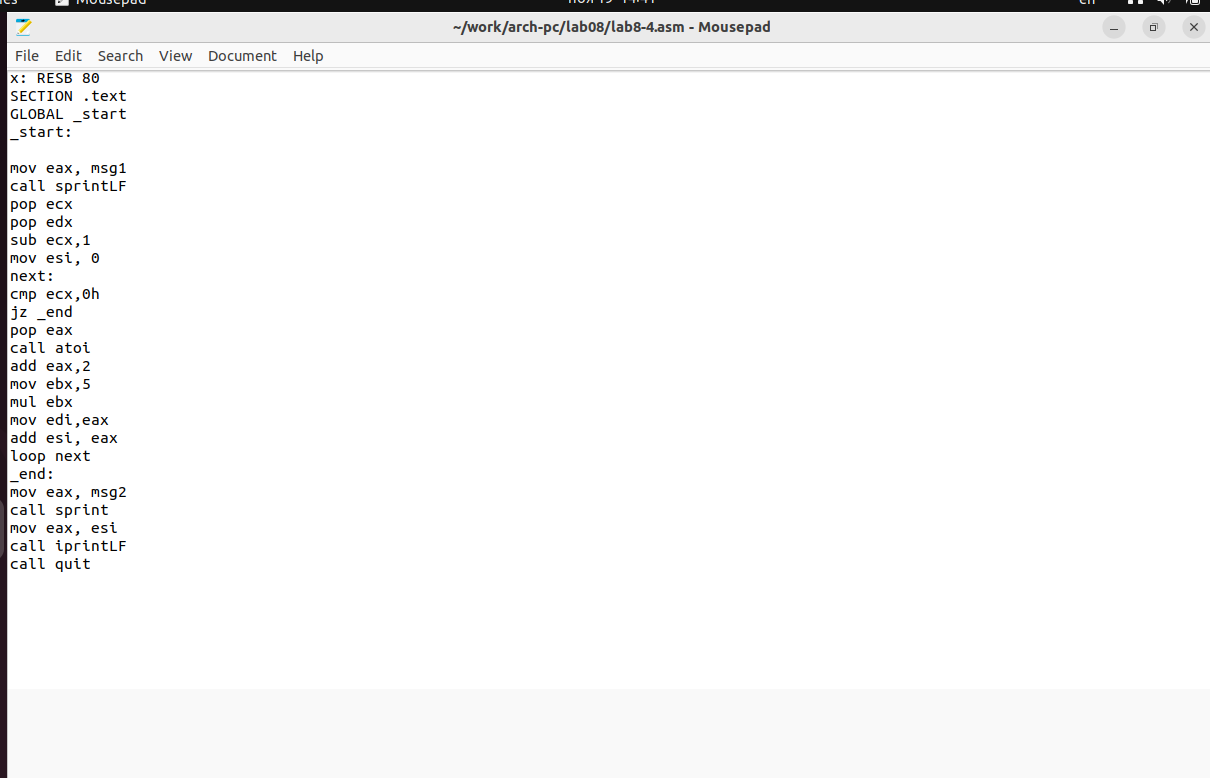
Задание: Напишите программу, которая находит сумму значений функции F(x) для х = x1, x2, …, xn, т.е. программа должна выводить значение f(x1) + f(x2) + … + f(xn). Значения xi передаются как аргументы. Вид функции f(x) выбрать из таблицы 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу на нескольких наборах x = x1, x2, …, xn.

Создаем файл lab8-4.asm с помощью утилиты touch (рис. [??]).

Создание файла

Создание файла

Открываем созданный файл для редактирования, вводим в него текст программы (рис. [??]).



Написание программы

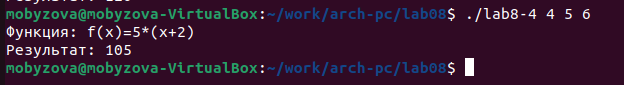
Далее создаем исполняемый файл и запускаем его (рис. [??], [??], [??]).

Создание исполняемого файла

Создание исполняемого файла

Запуск исполняемого файла

Запуск исполняемого файла



Запуск исполняемого файла

Посчитав аналитически сумму аргументов функции №10 из моего варианта, мы можем проверить правильность работы программы.

**Листинг №1. Изменённая программа lab8-3.asm**

%include 'in\_out.asm'  
SECTION .data  
msg db "Результат: ",0  
SECTION .text  
global \_start  
\_start:  
pop ecx  
pop edx  
sub ecx,1  
mov esi, 1  
next:  
cmp ecx,0h  
jz \_end  
pop eax  
call atoi  
imul esi, eax  
loop next  
\_end:  
mov eax, msg  
call sprint  
mov eax, esi  
call iprintLF  
call quit

**Листинг №2. Задание для самостоятельной работы**

%include 'in\_out.asm'  
SECTION .data  
msg1: DB 'Функция: f(x)=5\*(x+2) ',0  
msg2: DB 'Результат: ',0  
SECTION .bss  
x: RESB 80  
SECTION .text  
GLOBAL \_start  
\_start:  
mov eax, msg1  
call sprintLF  
pop ecx  
pop edx  
sub ecx,1  
mov esi, 0  
next:  
cmp ecx,0h  
jz \_end  
pop eax  
call atoi  
add eax,2  
mov ebx,5  
mul ebx  
mov edi,eax  
add esi, eax  
loop next  
\_end:  
mov eax, msg2  
call sprint  
mov eax, esi  
call iprintLF  
call quit

# 4 Выводы

В ходе лабораторной работы мы приобрели навыки написания программ с использованием циклов и обработкой аргументов командной строки.