Отчет по лабораторной работе №8

Программирование циклов. Обработка аргументов командной строки.

Пашутина Анна Алексеевна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Изучить работу циклов и обработкой аргументов командной строки.

2 Задание

Написать программы с использованием циклов и обработкой аргументов командной строки.

3 Выполнение лабораторной работы

3.1 Реализация циклов в NASM

Создаем каталог для программ лабораторной работы №8 с помощью команды mkdir , и в нем создаем файл lab8-1.asm с помощью команды touch(рис.1).

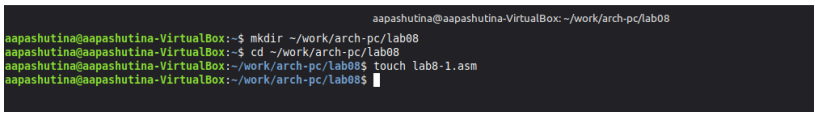

Рисунок 3.1: Создаем каталог и файл lab8-1.asm

Открываем файл в Midnight Commander с помощью команды mc и заполняем его в соответствии с листингом 8.1 (рис.2).

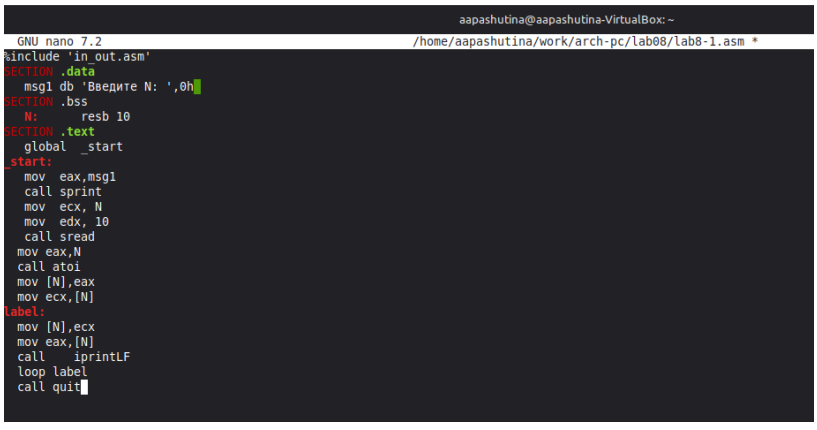

Рисунок 3.2: Заполняем файл в соответствии с заданием

Создаем исполняемый файл и запускаем его (рис.3).

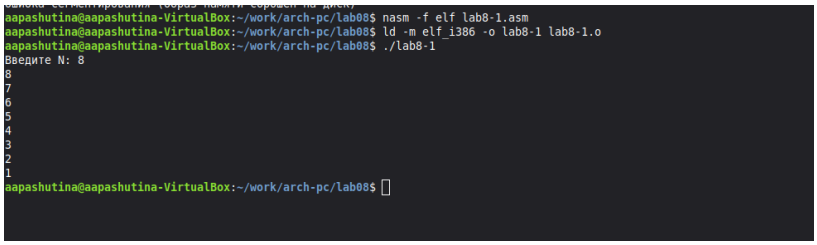

Рисунок 3.3: Запускаем файл и проверяем его работу

Снова открываем файл для редактирования и изменяем его, добавив изменение значения регистра в цикле (рис.4).

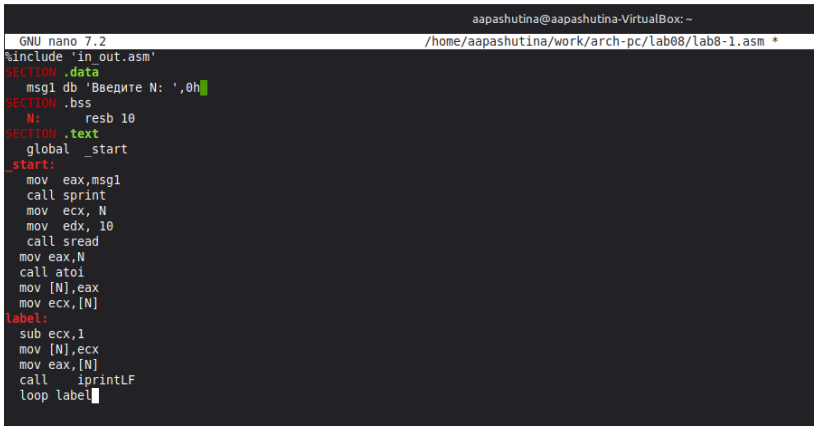

Рисунок 3.4: Изменяем файл в соответствии с заданием

Создаем исполняемый файл и запускаем его (рис.5).

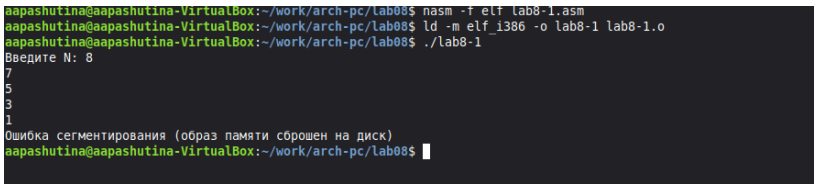

Рисунок 3.5: Запускаем файл и смотрим на его работу

Регистр ecx принимает значения 9,7,5,3,1(на вход подается число 10, в цикле label данный регистр уменьшается на 2 командой sub и loop).

Число проходов цикла не соответсвует числу N, так как уменьшается на 2.

Снова открываем файл для редактирования и изменяем его, чтобы все корректно работало (рис.6).

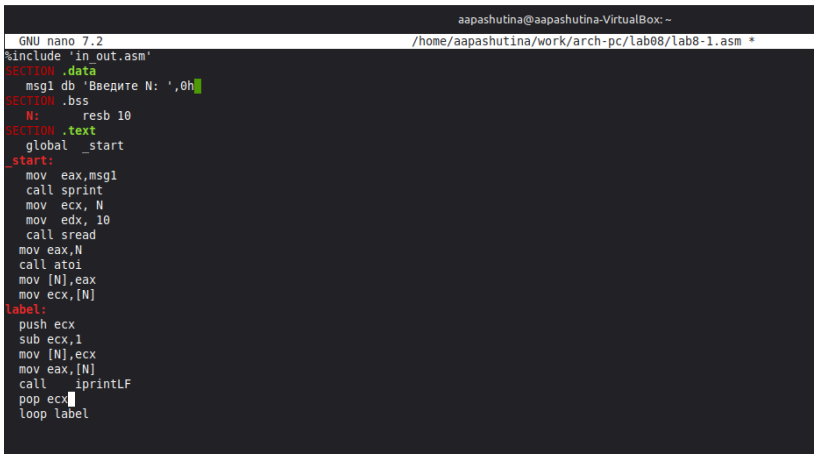

Рисунок 3.6: Редактируем файл

Создаем исполняемый файл и запускаем его (рис.7).

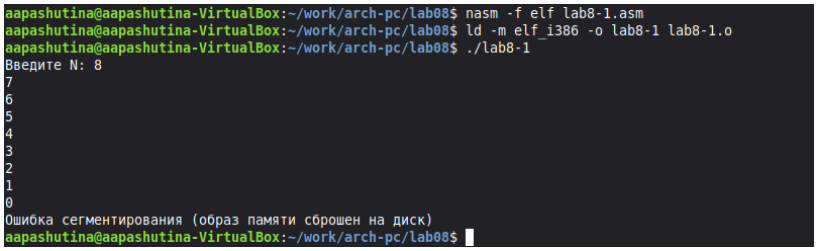

Рисунок 3.7: Проверяем, сошелся ли наш вывод с данным в условии выводом

В данном случае число проходов цикла равна числу N.

3.2 Обработка аргументов командной строки.

Создаем новый файл lab8-2.asm с помощью команды touch(рис.8).

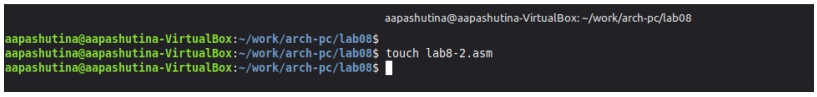

Рисунок 3.8: Создаем файл lab8-2.asm

Открываем файл в Midnight Commander с помощью команды mc и заполняем его в соответствии с листингом 8.2 (рис.9).

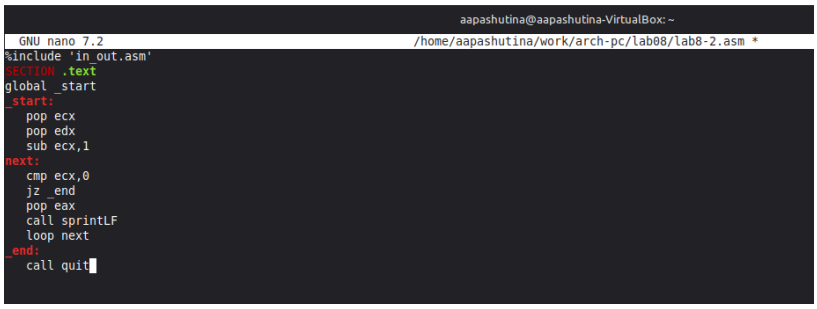

Рисунок 3.9: Заполняем файл в соответствии с заданием

Создаем исполняемый файл и проверяем его работу, указав аргументы (рис.10).

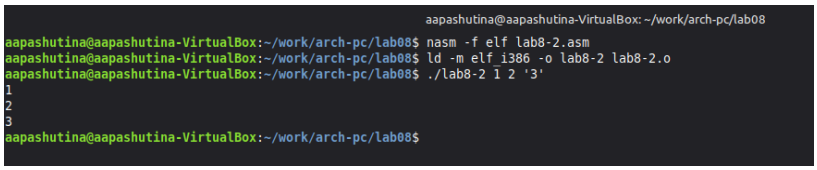

Рисунок 3.10: Смотрим на работу программ

Программой было обработано 3 аргумента.

Создаем новый файл lab8-3.asm с помощью команды touch(рис.11).

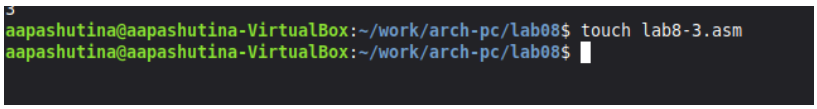

Рисунок 3.11: Создаем файл lab8-3.asm

Открываем файл и заполняем его в соответствии с листингом 8.3 (рис.12).

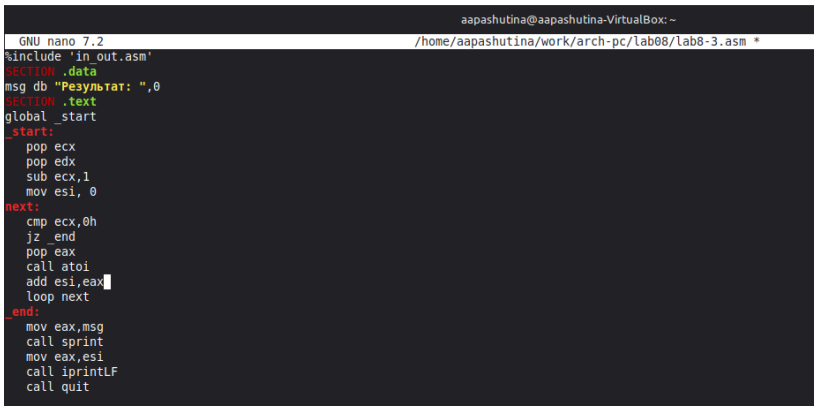

Рисунок 3.12: Заполняем файл в соответствии с заданием

Создаём исполняемый файл и запускаем его, указав аргументы (рис.13).

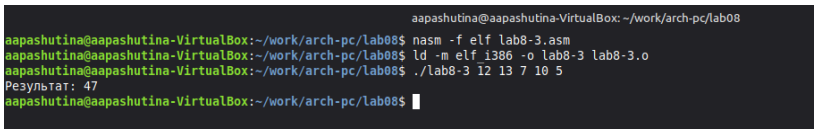

Рисунок 3.13: Смотрим на работу программы

Снова открываем файл для редактирования и изменяем его, чтобы вычислялось произведение вводимых значений (рис.14).

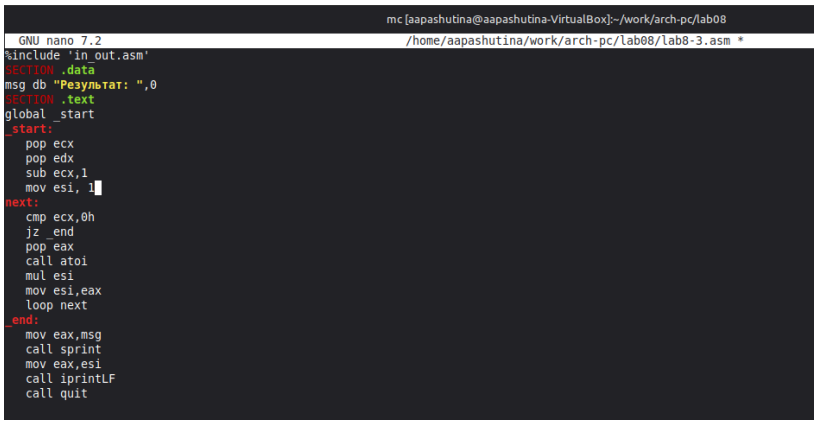

Рисунок 3.14: Изменяем файл

Создаём исполняемый файл и запускаем его, указав аргументы (рис.15).

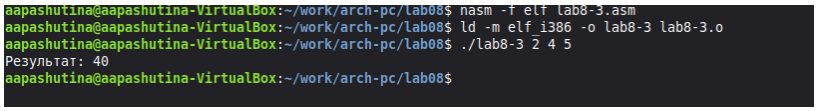

Рисунок 3.15: Проверяем работу файла(работает правильно)

3.3 Задание для самостоятельной работы

В лабораторной работе №6 программа высчитала мой номер варианта, который равен 3.

1. Напишите программу, которая находит сумму значений функции 𝑓(𝑥) для 𝑥 = 𝑥1, 𝑥2, ..., 𝑥𝑛, т.е. программа должна выводить значение 𝑓(𝑥1) + 𝑓(𝑥2) + ... + 𝑓(𝑥𝑛). Значения 𝑥𝑛 передаются как аргументы. Вид функции 𝑓(𝑥) выбрать из таблицы 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 6. Создайте исполняемый файл и проверьте его работу на нескольких наборах 𝑥 = 𝑥1, 𝑥2, ..., 𝑥𝑛.

Создаем новый файл lab8-4.asm с помощью команды touch(рис.16).

Рисунок 3.16: Создаем файл lab8-4.asm

Открываем его и пишем программу, которая выведет сумму значений, получившихся после решения выражения (10*x-5) (рис.17).

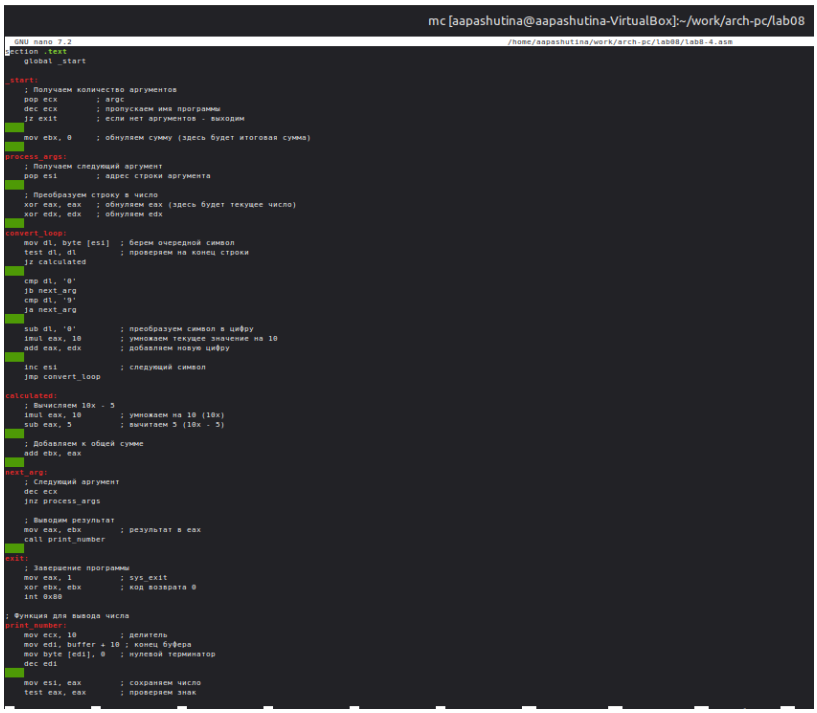

Рисунок 3.17: Пишем программу, которая выполнит задание

Транслируем файл и смотрим на работу программы (рис.18).

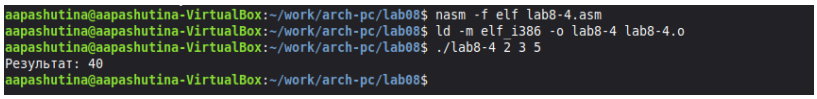

Рисунок 3.18: Смотрим на рабботу программы при x1=2 x2=3 x1=5(всё верно)

Транслируем файл и смотрим на работу программы (рис.19).

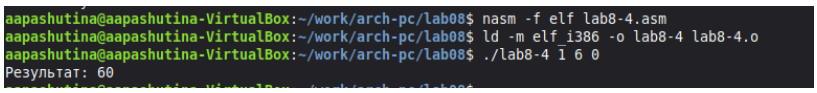

Рисунок 3.19: Смотрим на рабботу программы при x1=1 x2=6 x1=0(всё верно)

4 Выводы

Мы научились решать программы с использованием циклов и обработкой аргументов командной строки.