Отчет по лабораторной работе №8

Дисциплина: архитектура компьютера

Сокирка Анна Константиновна

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

# 2 Задание

1. Реализация циклом в NASM
2. Обработка аргументов командной строки
3. Самостоятельное написание программы по материалам лабораторной работы

# 3 Теоретическое введение

Стек — это структура данных, организованная по принципу LIFO («Last In — First Out» или «последним пришёл — первым ушёл»). Стек является частью архитектуры процессора и реализован на аппаратном уровне. Для работы со стеком в процессоре есть специальные регистры (ss, bp, sp) и команды. Основной функцией стека является функция сохранения адресов возврата и передачи аргументов при вызове процедур. Кроме того, в нём выделяется память для локальных переменных и могут временно храниться значения регистров.

# 4 Выполнение лабораторной работы

## 4.1 Реализация циклов в NASM

Создам каталог для программам лабораторной работы № 8, перейду в него и создам файл lab8-1.asm (рис. 1).



Рис. 1: Создание каталога

Введу в файл lab8-1.asm текст программы из листинга 8.1. Создам исполняемый файл и проверю его работу (рис. 2).



Рис. 2: Ввод программы из листинга

Изменю текст программы, добавив изменение значение регистра ecx в цикле. Создам исполняемый файл и проверю его работу (рис. 3).

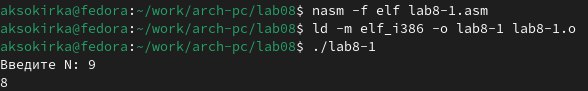


Рис. 3: Изменение текста программы

Из-за того, что теперь регистр ecx на каждой итерации уменьшается на 2 значения, количество итераций уменьшается вдвое. Число проходов цикла не соответсвует значению 𝑁 введенному с клавиатуры.

Внесу изменения в текст программы, добавив команды push и pop. Создам исполняемый файл и проверю его работу (рис. 4).

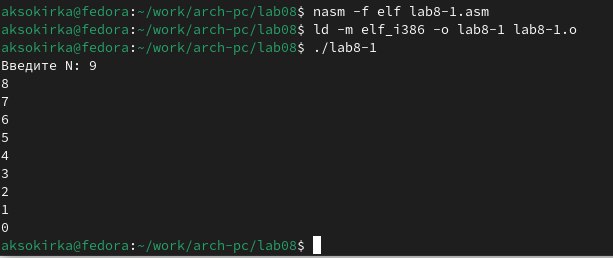


Рис. 4: Изменение текста программы

Теперь количество итераций совпадает введенному N, но произошло смещение выводимых чисел на -1

## 4.2 Обработка аргументов командной строки

Создам файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и введу в него текст программы из листинга 8.2. (рис. 5).

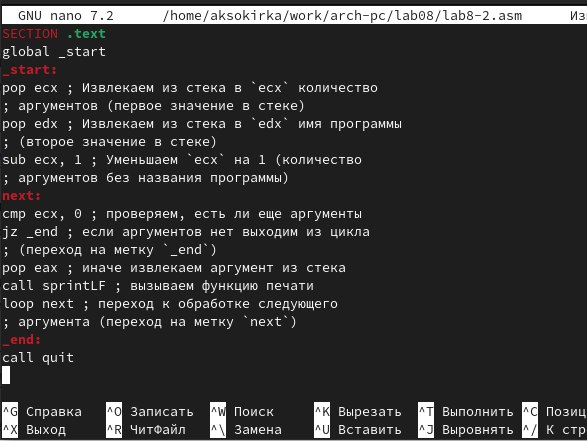


Рис. 5: Создание файла

Создам исполняемый файл и запущу его, указав аргументы (рис. 6).

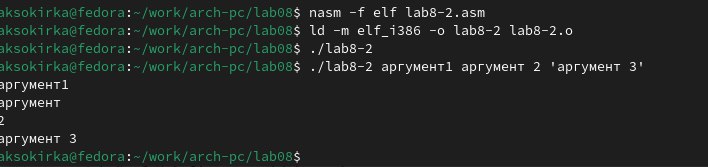


Рис. 6: Создание исполняемого файла

Программой было обратоно то же количество аргументов,которое было введено

Создам файл lab8-3.asm в каталоге ~/work/archpc/lab08 и введу в него текст программы из листинга 8.3. (рис. 7).

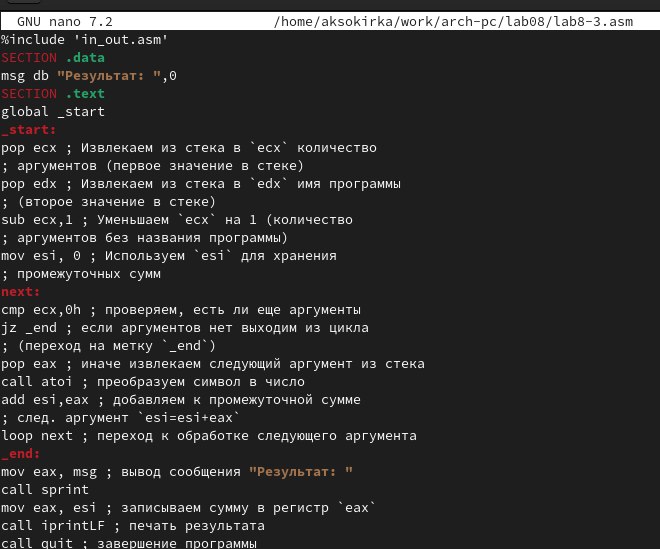


Рис. 7: Создание файла

Создайте исполняемый файл и запустите его, указав аргументы (рис. 8).

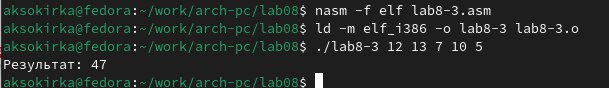


Рис. 8: Создание исполняемого файла

Изменяю программу так, чтобы указанные аргументы умножались (рис. 9).

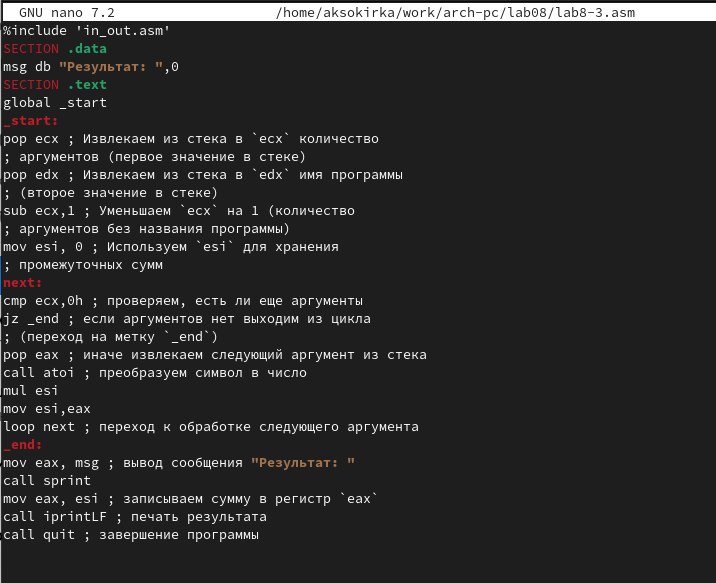


Рис. 9: Изменение текста программы

Запускаю программу (рис. 10).

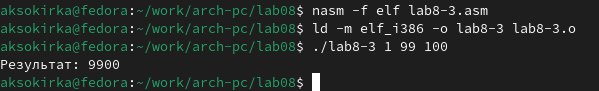


Рис. 10: Запуск программы

## 4.3 Задание для самостоятельной работы

Пишу программу, которая будет находить сумму значений для функции f(x) = 8x-3, которая совпадает с моим девятнадцатым вариантом (рис. 11).

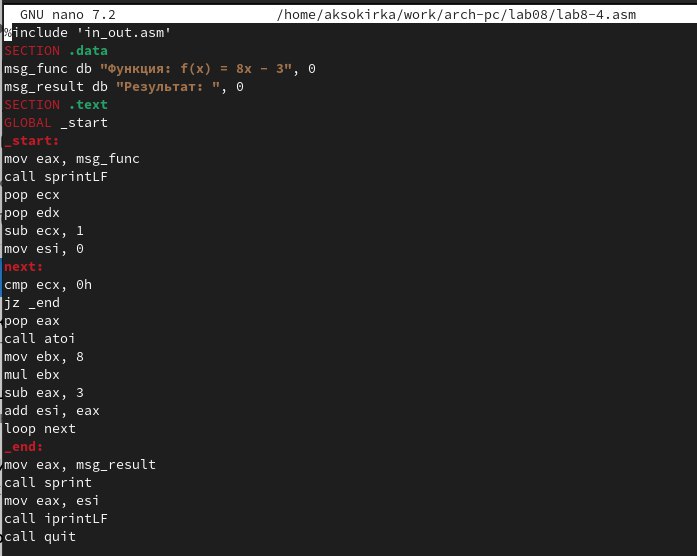


Рис. 11: Программа для нахождения суммы значений функции

Запускаю программу (рис. 12).

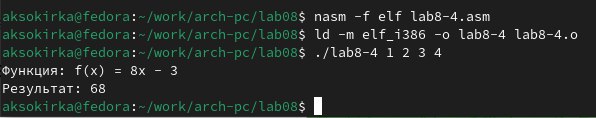


Рис. 12: Запуск программы

# 5 Выводы

В результате выполнения данной лабораторной работы я приобрела навыки написания программ с использованием циклов, а также научилась обрабатывать аргументы командной строки.

# 6 Список литературы

https://esystem.rudn.ru/pluginfile.php/2089095/mod\_resource/content/0/Лабораторная%20работа%20№8.%20Программирование%20цикла.%20Обработка%20аргументов%20командной%20строки..pdf