Отчёт по лабораторной работе №8

Дисциплина: Архитектура компьютера

Будник Александра Олеговна

Содержание

# 1 Цель работы

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

# 2 Выполнение лабораторной работы

## 2.1 Реализация циклов в NASM

Создала каталог для программам лабораторной работы № 8 и файл lab8-1.asm (рис. [[1](#fig:001)])

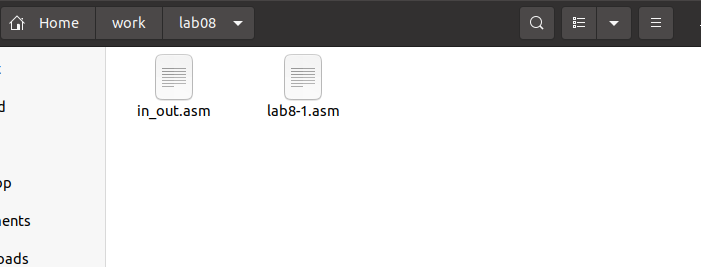


Figure 1: Создан каталог

При реализации циклов в NASM с использованием инструкции loop необходимо помнить о том, что эта инструкция использует регистр ecx в качестве счетчика и на каждом шаге уменьшает его значение на единицу. В качестве примера рассмотрим программу, которая выводит значение регистра ecx.

Написала в файл lab8-1.asm текст программы из листинга 8.1. (рис. [[2](#fig:002)]) Создала исполняемый файл и проверила его работу. (рис. [[3](#fig:003)])

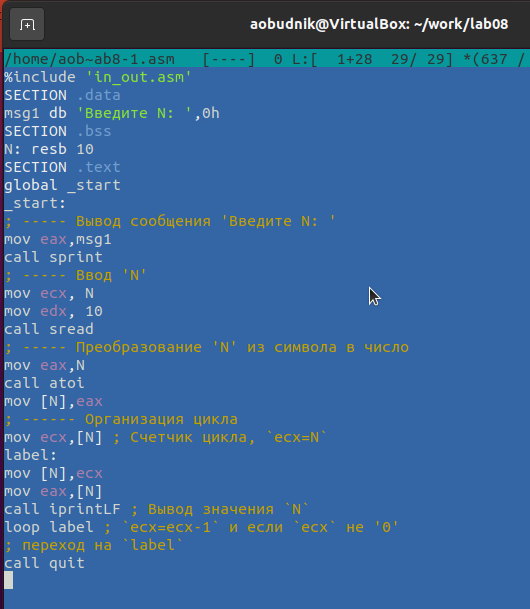


Figure 2: Программа lab8-1.asm

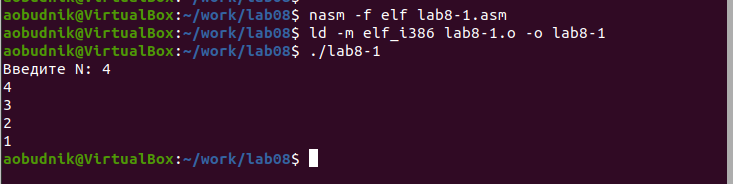


Figure 3: Запуск программы lab8-1.asm

Данный пример показывает, что использование регистра ecx в теле цилка loop может привести к некорректной работе программы. Изменяю текст программы, добавив изменение значение регистра ecx в цикле. (рис. [[4](#fig:004)]) Программа запускает бесконечный цикл при нечетном N и выводит только нечетные числа при четном N. (рис. [[5](#fig:005)])

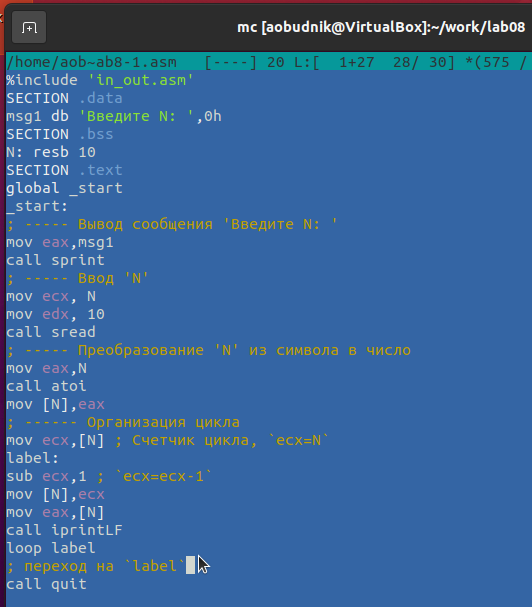


Figure 4: Программа lab8-1.asm

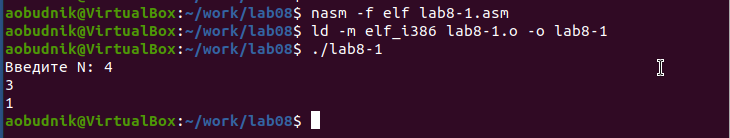


Figure 5: Запуск программы lab8-1.asm

Для использования регистра ecx в цикле и сохранения корректности работы программы можно использовать стек. Внес изменения в текст программы добавив команды push и pop (добавления в стек и извлечения из стека) для сохранения значения счетчика цикла loop. (рис. [[6](#fig:006)]) Создаю исполняемый файл и проверяю его работу. (рис. [[7](#fig:007)]) Программа выводит числа от N-1 до 0, число проходов цикла соответсвует N.

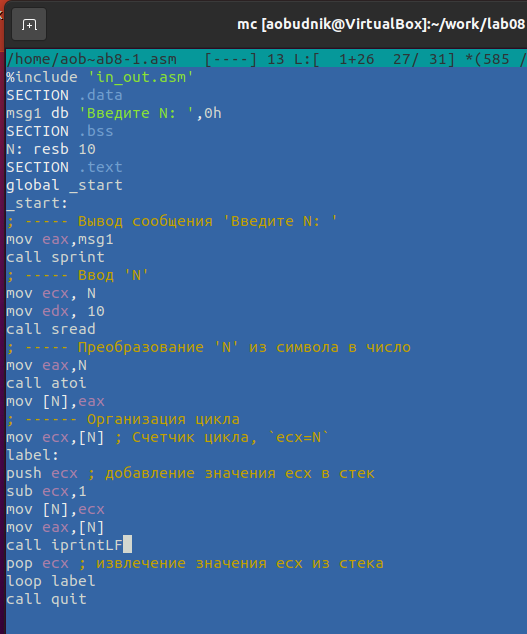


Figure 6: Программа lab8-1.asm

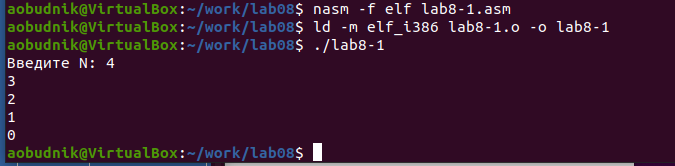


Figure 7: Запуск программы lab8-1.asm

Создала файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и написала в него текст программы из листинга 8.2. (рис. [[8](#fig:008)]) Компилирую исполняемый файл и запускаю его, указав аргументы. Программа обработала 5 аргументов. Аргументами считаются слова/числа, разделенные пробелом. (рис. [[9](#fig:009)])

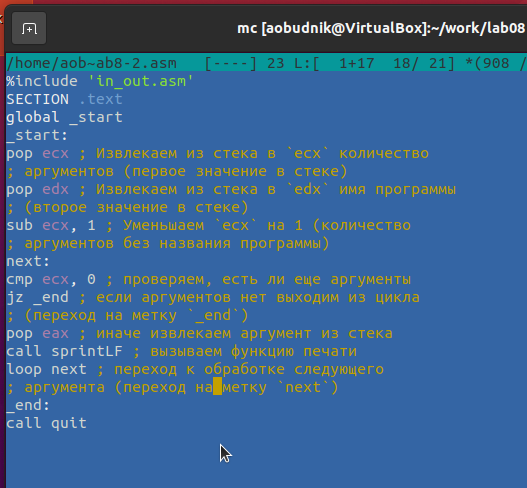


Figure 8: Программа lab8-2.asm

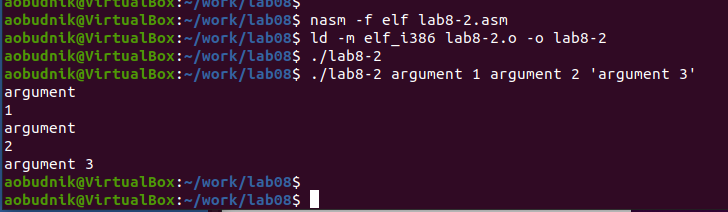


Figure 9: Запуск программы lab8-2.asm

Рассмотрим еще один пример программы, которая выводит сумму чисел, которые передаются в программу как аргументы. (рис. [[10](#fig:010)]) (рис. [[11](#fig:011)])

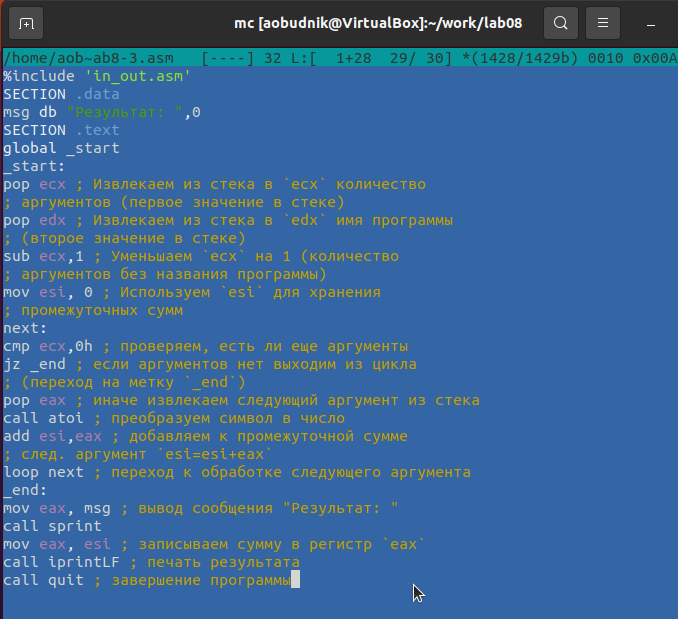


Figure 10: Программа lab8-3.asm

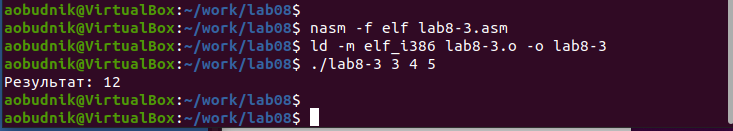


Figure 11: Запуск программы lab8-3.asm

Изменю текст программы из листинга 8.3 для вычисления произведения аргументов командной строки. (рис. [[12](#fig:012)]) (рис. [[13](#fig:013)])

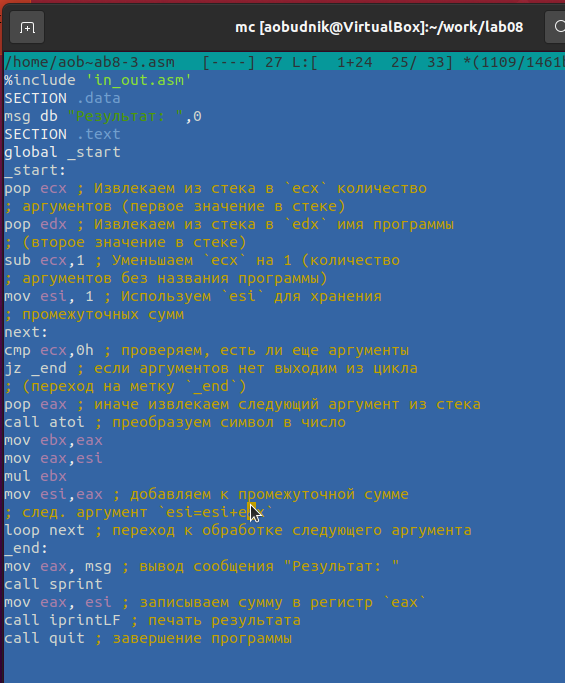


Figure 12: Программа lab8-3.asm

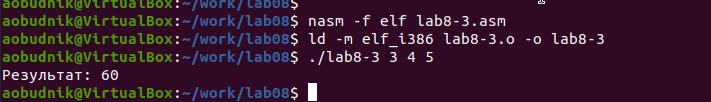


Figure 13: Запуск программы lab8-3.asm

## 2.2 Самостоятельное задание

Напишите программу, которая находит сумму значений функции для , т.е. программа должна выводить значение . Значения передаются как аргументы. Вид функции 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 .(рис. [[14](#fig:014)]) (рис. [[15](#fig:015)])

для варианта 17



Figure 14: Программа lab8-4.asm

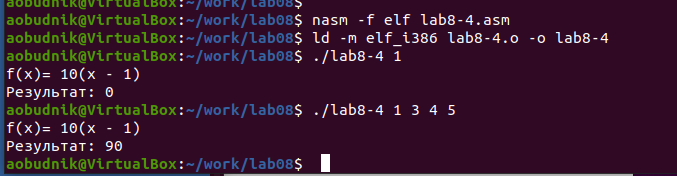


Figure 15: Запуск программы lab8-4.asm

Убедились, что программа считает правильно .

# 3 Выводы

Освоили работы со стеком, циклом и аргументами на ассемблере nasm.