Отчёт по лабораторной работе 8

Архитектура компьютеров

Сюй Хайфэн

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

Создал каталог для программ лабораторной работы № 8 и файл lab8-1.asm (рис. 1).

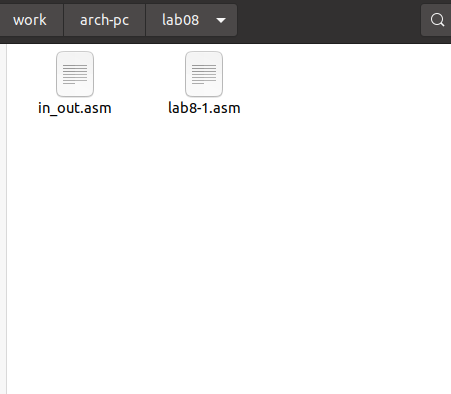


Рис. 1: Создание каталога

При реализации циклов в NASM с использованием инструкции loop важно учитывать, 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.

Написал в файл lab8-1.asm текст программы из листинга 8.1 (рис. 2). Создал исполняемый файл и проверил его работу (рис. 3).

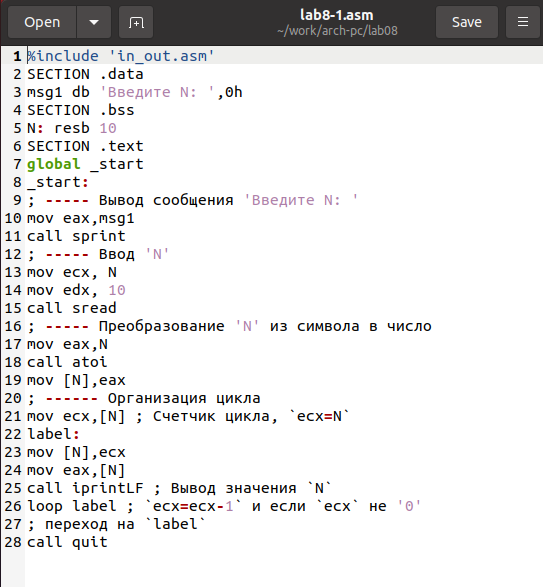


Рис. 2: Программа lab8-1.asm

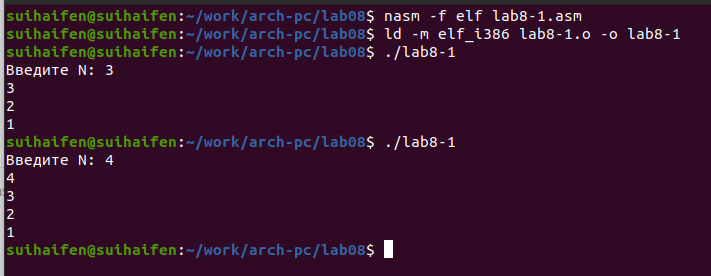


Рис. 3: Запуск программы lab8-1.asm

Этот пример демонстрирует, что использование регистра ecx в теле цикла loop может привести к некорректной работе программы. Изменил текст программы, добавив изменение значения регистра ecx в цикле (рис. 4). Теперь программа запускает бесконечный цикл при нечетном N и выводит только нечетные числа при четном N (рис. 5).

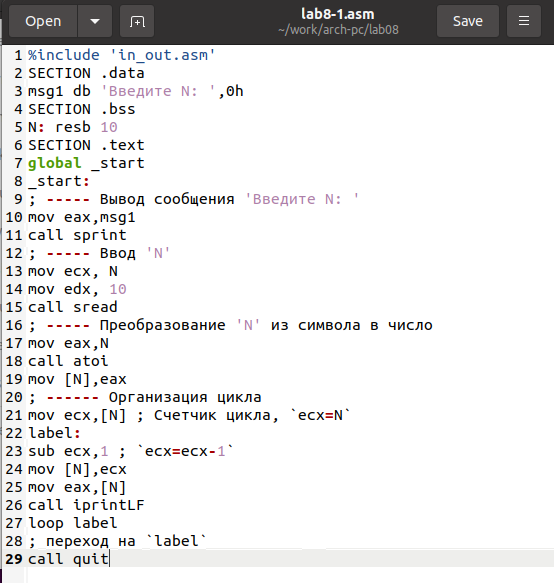


Рис. 4: Измененная программа lab8-1.asm

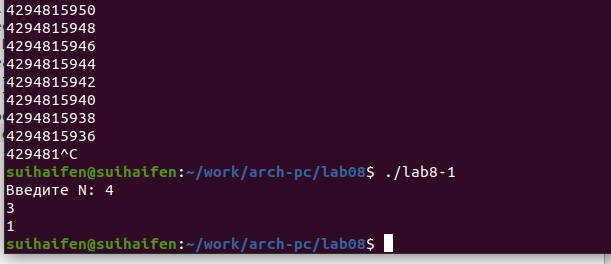


Рис. 5: Запуск измененной программы lab8-1.asm

Для корректной работы программы с регистром ecx в цикле использовал стек. Внес изменения в текст программы, добавив команды push и pop (для добавления в стек и извлечения из него значений), чтобы сохранить значение счетчика цикла loop (рис. 6). Создал исполняемый файл и проверил его работу (рис. 7). Теперь программа выводит числа от N-1 до 0, при этом число проходов цикла соответствует значению N.

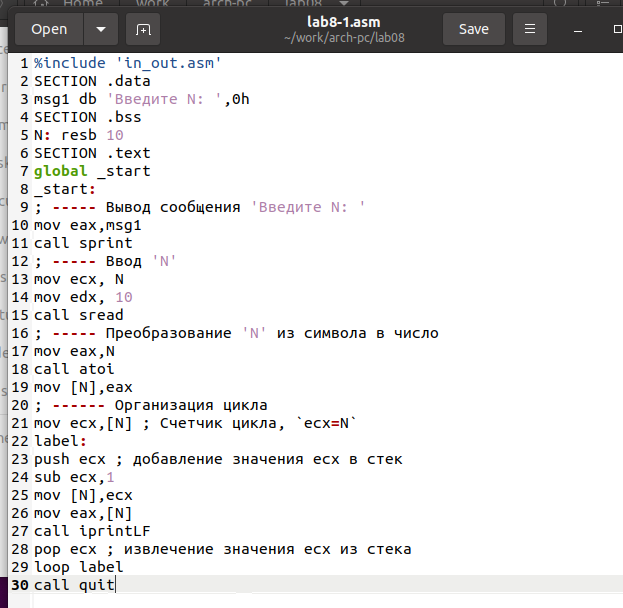


Рис. 6: Исправленная программа lab8-1.asm

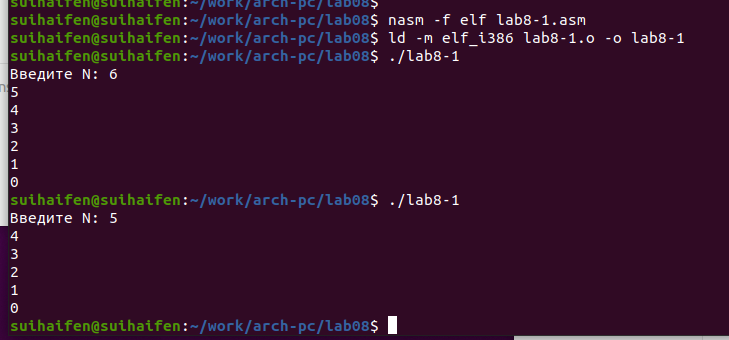


Рис. 7: Запуск исправленной программы lab8-1.asm

Создал файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и написал в него текст программы из листинга 8.2 (рис. 8). Компилирую исполняемый файл и запускаю его, указав аргументы. Программа обработала 4 аргумента. Аргументами считаются слова/числа, разделенные пробелом (рис. 9).

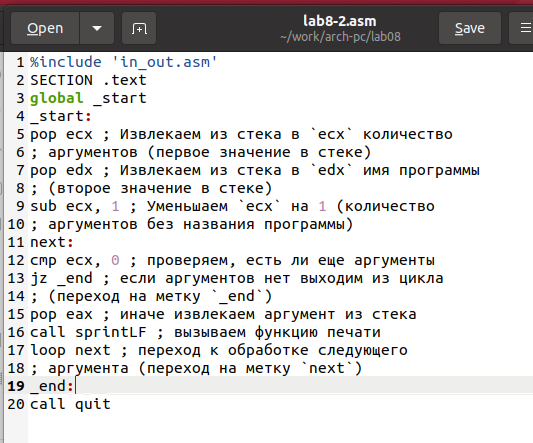


Рис. 8: Программа lab8-2.asm

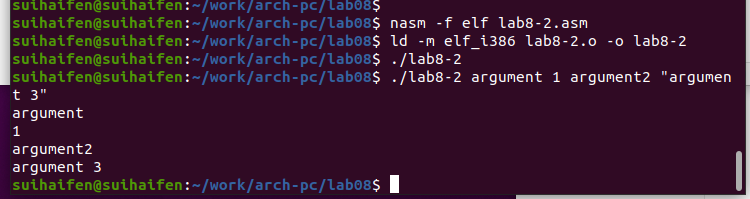


Рис. 9: Запуск программы lab8-2.asm

Рассмотрим еще один пример программы, которая выводит сумму чисел, передаваемых в программу как аргументы (рис. 10) (рис. 11).

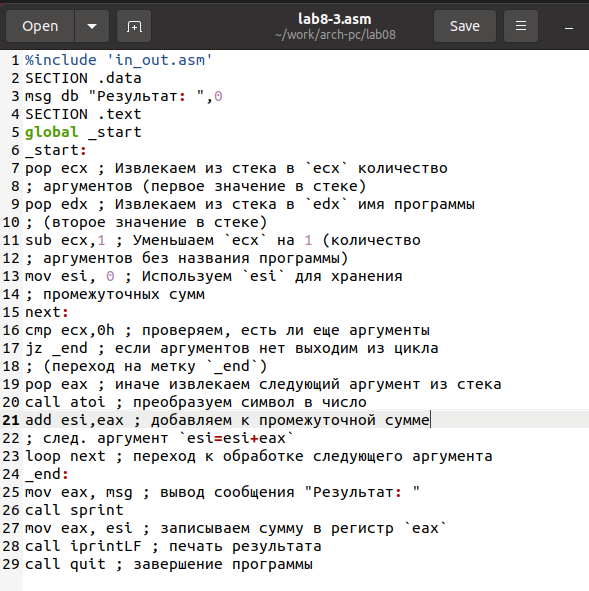


Рис. 10: Программа lab8-3.asm

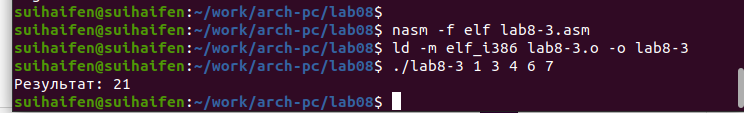


Рис. 11: Запуск программы lab8-3.asm

Изменил текст программы из листинга 8.3 для вычисления произведения аргументов командной строки (рис. 12) (рис. 13).

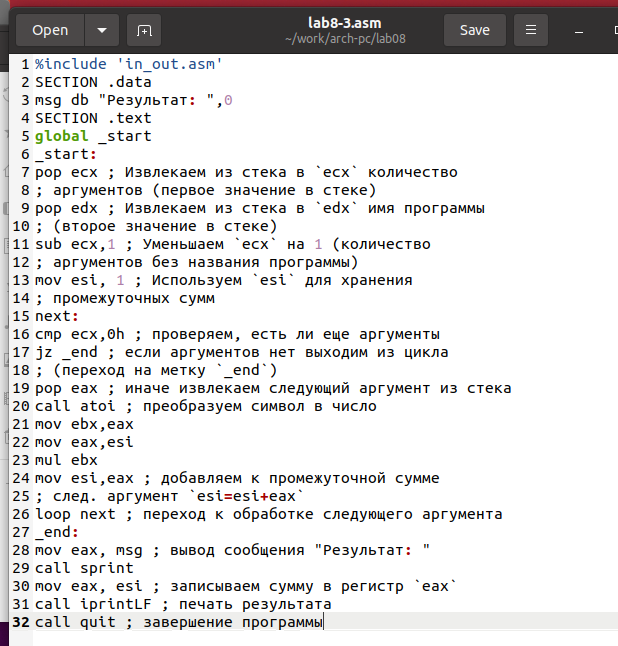


Рис. 12: Программа lab8-3.asm

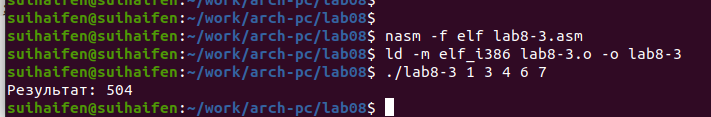


Рис. 13: Запуск программы lab8-3.asm

## 2.2 Самостоятельное задание

Написать программу, которая находит сумму значений функции для , т.е. программа должна выводить значение . Значения передаются как аргументы. Вид функции следует выбрать согласно таблице 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создал исполняемый файл и проверил его работу на нескольких наборах (рис. 14) (рис. 15).

Для варианта 9:

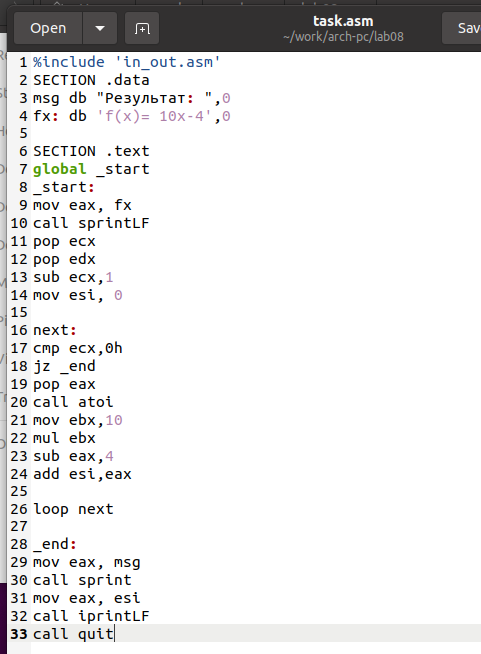


Рис. 14: Программа task.asm

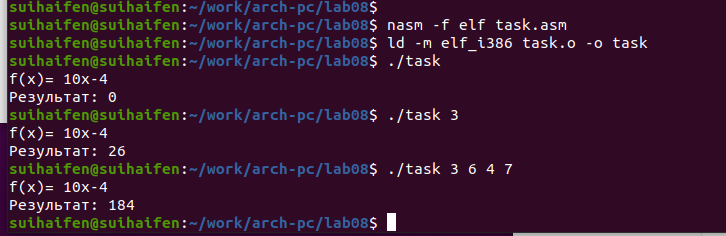


Рис. 15: Запуск программы task.asm

Убедился, что программа правильно вычисляет .

# 3 Выводы

Освоила работу со стеком, циклами и аргументами на ассемблере NASM.