Отчёт по лабораторной работе № 9

НММБД-02-22

Паулу Антонью Жоау

Содержание

# 1 Цель работы

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

# 2 Задание

1. Реализацовать циклы в NASM
2. Выполнить обработку аргументов командной строки
3. Выполнить задание для самостоятельной работы

# 3 Выполнение лабораторной работы

## 3.1 Реализация циклов в NASM

Создали каталог для программам лабораторной работы № 9, перейшли в него и создали файл lab9-1.asm: (рис. 1)

Рис. 1: lab9-1.asm

Рис. 1: lab9-1.asm

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 рассмотрели программу, которая выводит значение регистра ecx. Внимательно изучили текст программы (Листинг 9.1).

Ввели в файл lab9-1.asm текст программы из листинга 9.1. (рис. 2) Создали исполняемый файл и проверили его работу. (рис. 3)

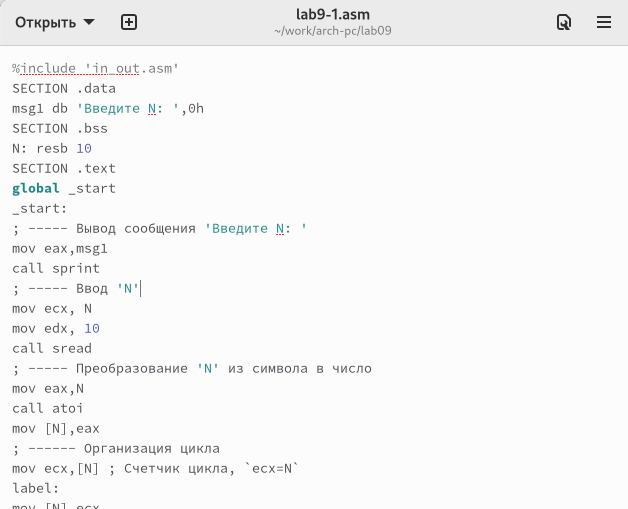


Рис. 2: Текст программы

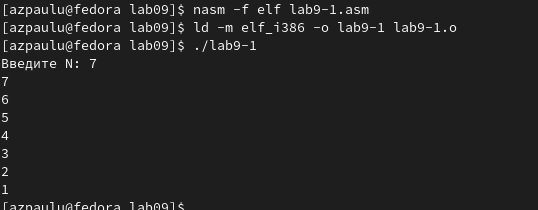


Рис. 3: Исполняемый файл

Данный пример показывает, что использование регистра ecx в теле цилка loop может привести к некорректной работе программы. Изменили текст программы добавив изменение значения регистра ecx в цикле: (рис. 4)

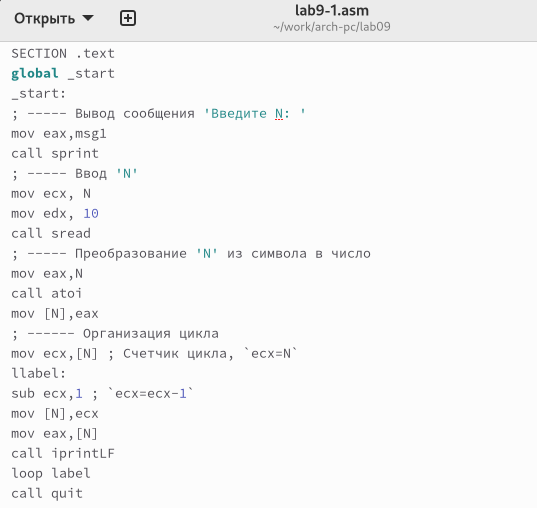


Рис. 4: Текст программы

Создали исполняемый файл и проверили его работу. (рис. 5) Регистр ecx принимает следующие значения в цикле: (рис. 6). Число проходов цикла не соответствует значению N введенному с клавиатуры.

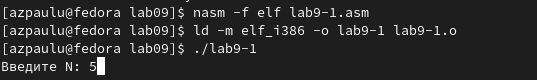


Рис. 5: Исполняемый файл

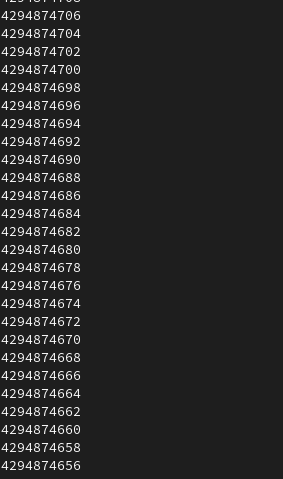


Рис. 6: Значения в цикле

Для использования регистра ecx в цикле и сохранения корректности работы программы можно использовать стек. Внесли изменения в текст программы добавив команды push и pop (добавления в стек и извлечения из стека) для сохранения значения счетчика цикла loop: (рис. 7)

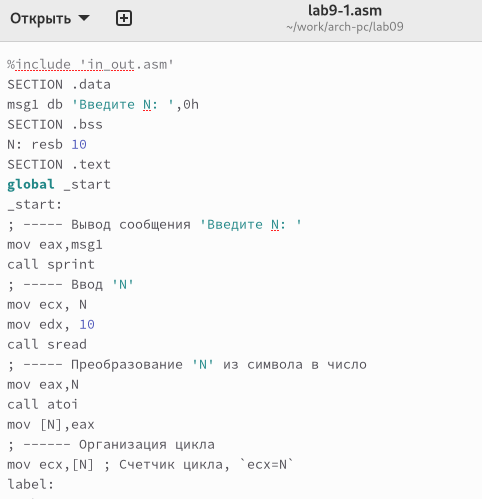


Рис. 7: Текст программы

Создали исполняемый файл и проверили его работу. (рис. 8) В данном случае число проходов цикла соответствует значению N введенному с клавиатуры.

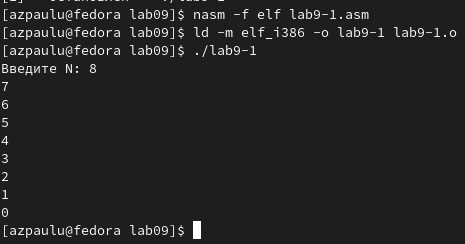


Рис. 8: Исполняемый файл

## 3.2 Обработка аргументов командной строки

При разработке программ иногда встает необходимость указывать аргументы, которые будут использоваться в программе, непосредственно из командной строки при запуске программы. При запуске программы в NASM аргументы командной строки загружаются в стек в обратном порядке, кроме того в стек записывается имя программы и общее количество аргументов. Последние два элемента стека для программы, скомпилированной NASM, – это всегда имя программы и количество переданных аргументов. Таким образом, для того чтобы использовать аргументы в программе, их просто нужно извлечь из стека. Обработку аргументов нужно проводить в цикле. Т.е. сначала нужно извлечь из стека количество аргументов, а затем циклично для каждого аргумента выполнить логику программы. В качестве примера рассмотрели программу, которая выводит на экран аргументы командной строки. Внимательно изучили текст программы (Листинг 9.2).

Создали файл lab9-2.asm в каталоге ~/work/arch-pc/lab09 и ввели в него текст программы из листинга 9.2. (рис. 9), (рис. 10) Создали исполняемый файл и запустили его, указав аргументы: (рис. 11)

Рис. 9: lab9-2.asm

Рис. 9: lab9-2.asm

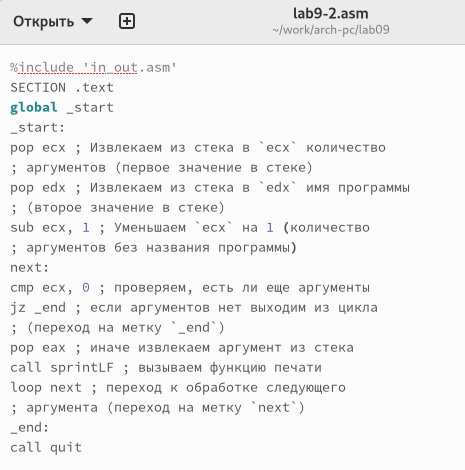


Рис. 10: Текст программы

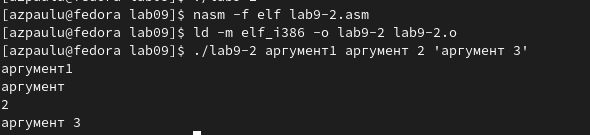


Рис. 11: Исполняемый файл

Четыре аргумента было обработано программой. Рассмотрели еще один пример программы, которая выводит сумму чисел, которые передаются в программу как аргументы. Создали файл lab9-3.asm в каталоге ~/work/arch-pc/lab09 и ввели в него текст программы из листинга 9.3. (рис. 12), (рис. 13)

Рис. 12: lab9-3.asm

Рис. 12: lab9-3.asm

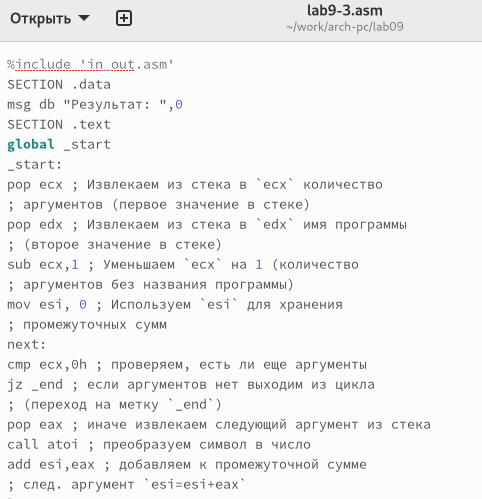


Рис. 13: Текст программы

Создали исполняемый файл и запустили его, указав аргументы. (рис. 14)

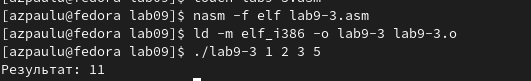


Рис. 14: Исполняемый файл

Изменили текст программы из листинга 9.3 для вычисления произведения аргументов командной строки. (рис. 15), (рис. 16)

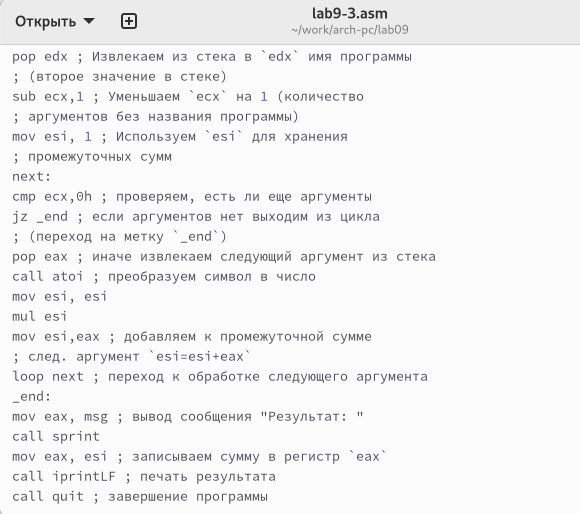


Рис. 15: Текст программы

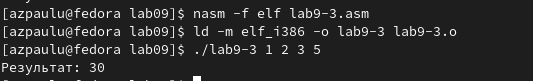


Рис. 16: Исполняемый файл

## 3.3 Задание для самостоятельной работы

Написали 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 xi передаются как аргументы. Вид функции f(x) выбрали из таблицы 9.1 вариантов заданий в соответствии с вариантом 13, полученным при выполнении лабораторной работы № 7. Создали исполняемый файл и проверили его работу на нескольких наборах. (рис. 17), (рис. 18)

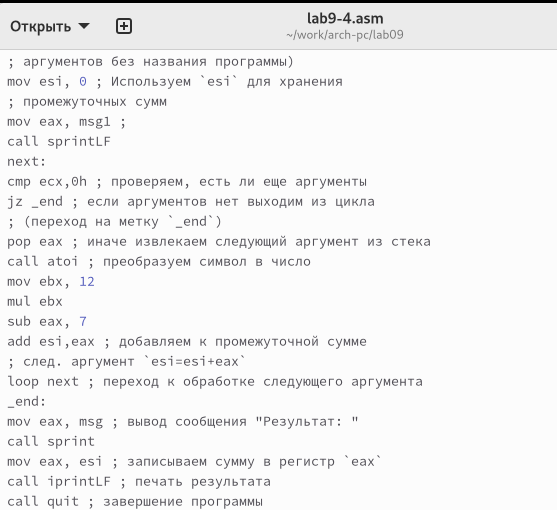


Рис. 17: Текст программы

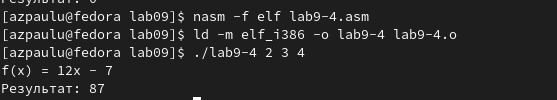


Рис. 18: Исполняемый файл

# 4 Выводы

В ходе выполнения лабораторной работы были приобретены навыки написания программ с использованием циклов и обработкой аргументов командной строки.