Лабораторная работа №9

Программирование цикла. Обработка аргументов командной строки.

Татьяна Алексеевна Коннова, НПИбд-01-22

Содержание

[1 Цель работы 1](#__RefHeading___Toc328_2569422953)

[1.1 Задание 1](#__RefHeading___Toc330_2569422953)

[1.2 Выполнение лабораторной работы 1](#__RefHeading___Toc332_2569422953)

[1.2.1 Реализация циклов в NASM 1](#__RefHeading___Toc334_2569422953)

[1.3 9.3.2. Обработка аргументов командной строки 9](#__RefHeading___Toc336_2569422953)

[1.4 Самостоятельная работа 15](#__RefHeading___Toc338_2569422953)

[2 Выводы 17](#__RefHeading___Toc340_2569422953)

# 1 Цель работы

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

## 1.1 Задание

Программирование ветвлений

## 1.2 Выполнение лабораторной работы

### 1.2.1 Реализация циклов в NASM

Создаем файл lab9-1.asm touch 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 рассмотрим программу (Листинг 9.1). Введем в файл lab9-1.asm текст программы из листинга 9.1. Создадим исполняемый файл и проверим его работу. Данный пример показывает, что использование регистра ecx в теле цикла loop может привести к некорректной работе программы. (рис. 1)  
(рис. 2)

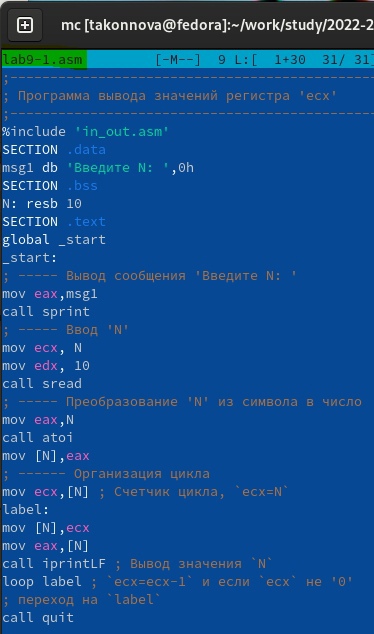


Рис. 1: lab9\_1

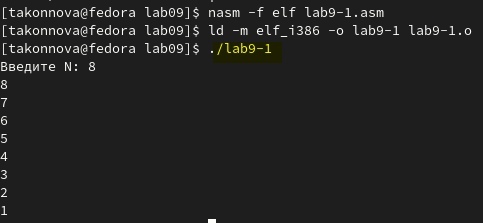


Рис. 2: lab9\_1

Изменим текст программы, добавив изменение значение регистра ecx в цикле: label:…

(рис. 3)

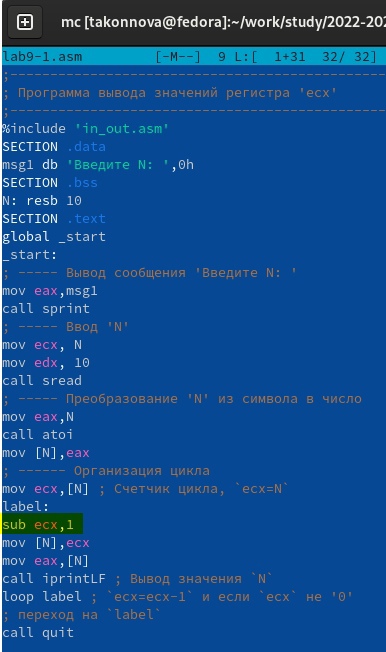


Рис. 3: lab9\_1

Создаем исполняемый файл и проверяем его работу.

(рис. 4)

(рис. 5)

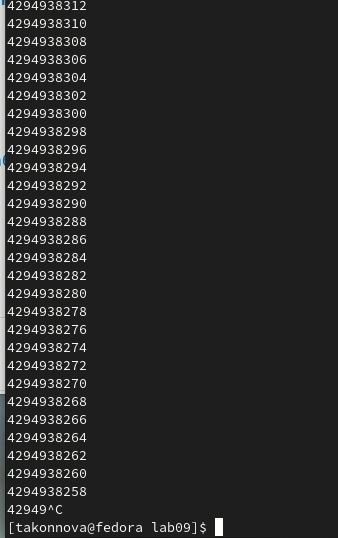


Рис. 4: lab9\_1

При четных:(создала еще один новый файл lab9\_1.asm, так как забыла про отделение четных чисел и нечетных)

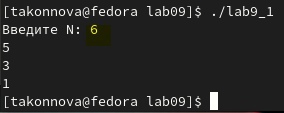


Рис. 5: lab9\_1

Соответствует ли число проходов цикла значению N введенному с клавиатуры?

* Ответ: бесконечный цикл выводится при нечетных N и нечетные числа выводятся при четных значениях N

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

Создаем исполняемый файл и проверяем его работу. (рис. 6) (рис. 7) Соответствует ли в данном случае число проходов цикла значению N введенному с клавиатуры?

* Ответ: программа обрабатывает, выводит числа от N-1 до 0, поэтому число проходов цикла есть само число N раз.

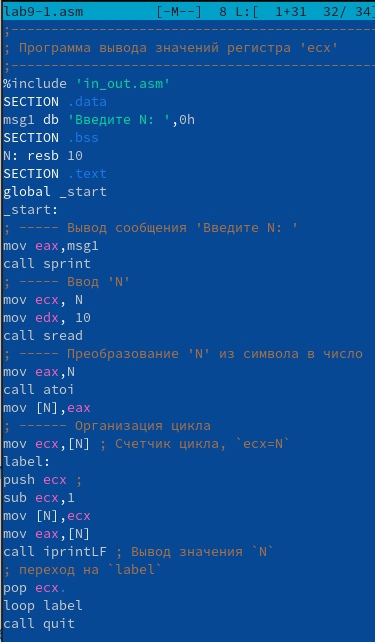


Рис. 6: lab9\_1

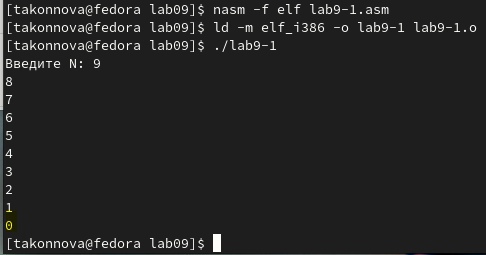


Рис. 7: lab9\_1

## 1.3 9.3.2. 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 передан- ных аргументов. Таким образом, для того чтобы использовать аргументы в программе, их про- сто 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 рас- смотрим программу, которая выводит на экран аргументы командной строки. Внимательно изучим текст программы (Листинг 9.2).

Создаем файл lab9-2.asm в каталоге ~/work/arch-pc/lab09 и введите в него текст программы из листинга 9.2. Создаем исполняемый файл и запускаем его, указав аргументы (рис. 8) (рис. 9)

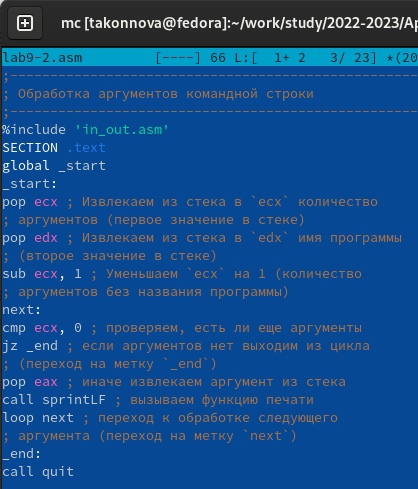


Рис. 8: lab9\_2

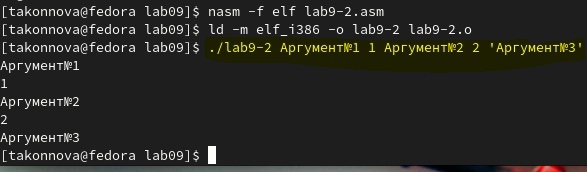


Рис. 9: lab9\_2

Сколько аргументов было обработано программой?  
\* Ответ: 5

Рассмотрим еще один пример программы которая выводит сумму чисел, которые передаются в программу как аргументы. Создайте файл lab9-3.asm, введите в него текст программы из листинга 9.3. Создайте исполняемый файл и запустите его, указав аргументы. (рис. 10) (рис. 11)

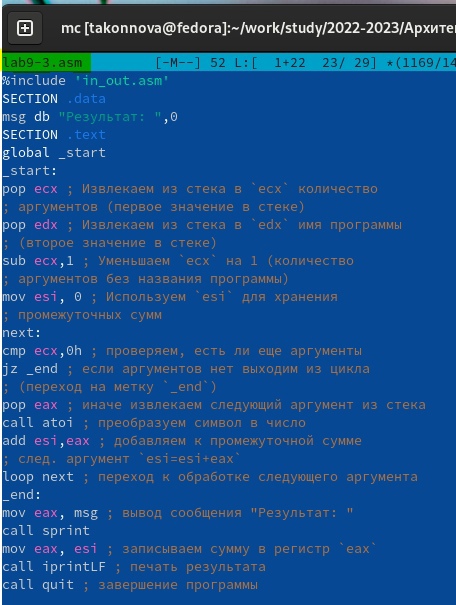


Рис. 10: lab9\_3

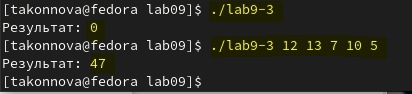


Рис. 11: lab9\_3

Изменим текст программы из листинга 9.3 для вычисления произведения аргументов командной строки.

(рис. 12) (рис. 13)

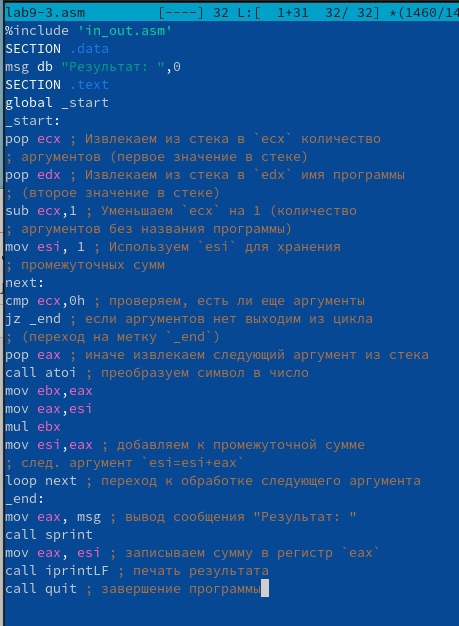


Рис. 12: lab9\_3

Рис. 13: lab9_3

Рис. 13: lab9\_3

## 1.4 Самостоятельная работа

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 xi передаются как аргументы. Вид функции f(x) выбрать из таблицы 9.1 вариантов заданий в соответствии с вариантом, по- лученным при выполнении лабораторной работы No 7. Создайте исполня- емый файл и проверьте его работу на нескольких наборах x = x1, x2, …, xn.

Вариант моей самостоятельной работы = №15, а именно f(x) = 6x + 13

(рис. 14) (рис. 15)

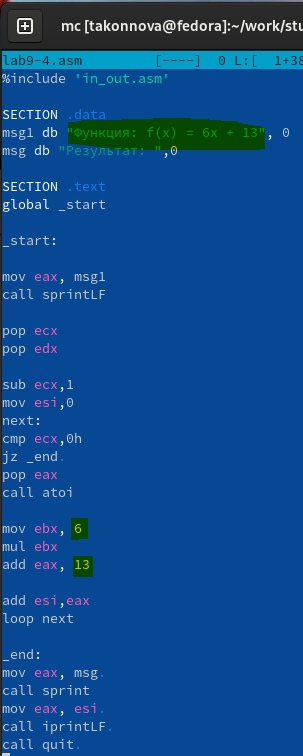


Рис. 14: samost\_rab\_lab9\_4

Рис. 15: samost_rab_lab9_4

Рис. 15: samost\_rab\_lab9\_4

# 2 Выводы

Мы приобрели навыки написания программ с использованием циклов и обработкой аргументов командной строки.