Лабораторная работа №8

Программирование цикла. Обработка аргументов командной строки

Налобин Михаил Дмитриевич

Содержание

# 1 Цель работы

Освоить навык работы с циклами и обработкой аргументов командой строки в программах на языке ассемблера NASM.

# 2 Ход работы

Создали каталог lab08 для файлов лабораторной работы №8 и в нем файл lab8-1.asm (рис. [1](#fig:001)).

Figure 1: Создание каталога lab08

Figure 1: Создание каталога lab08

Переписали в него текст из Листинга 8.1. (рис. [2](#fig:002)).

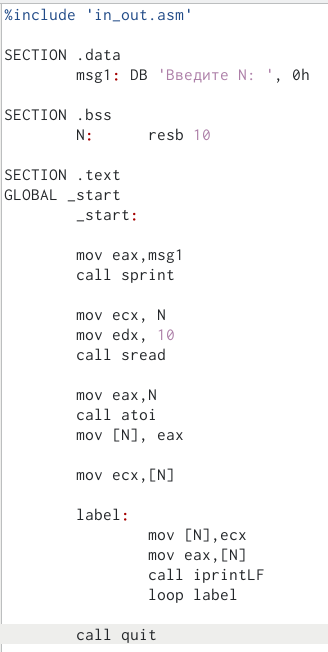


Figure 2: Код программы lab8-1.asm

После чего создали исполняемый файл и запустили его (рис. [3](#fig:003)).

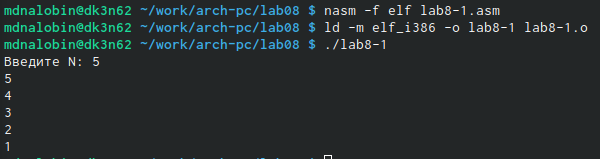


Figure 3: Запуск программы lab8-1

Далее изменяем содержание lab8-1.asm для демонстрации некорректной работы программы при использовании регистра ecx (рис. [4](#fig:004)).

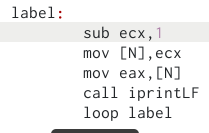


Figure 4: Измененная часть кода программы lab8-1.asm

Так же создали исполняемый файл и запустили его. В данной программе ecx принимает все значения, но до eax доходит половина, поэтому и число проходов сокращается вдвое(причем округление при получение десятичного числа в большую сторону), что мы видим при выводе, так как ecx за ход цикла убывает на 1 два раза (при sub и loop) (рис. [5](#fig:005)).

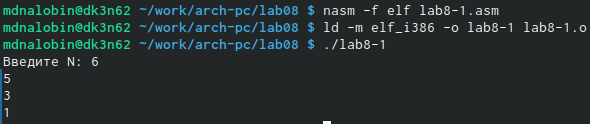


Figure 5: Запуск измененной программы lab8-1

Затем снова изменили код программы lab8-1.asm, чтобы использовать регистр ecx и при этом сохранить корректную работу цикла, использовав стек (рис. [6](#fig:006)).

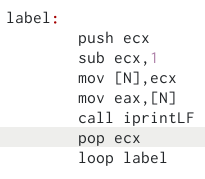


Figure 6: Повторно измененная часть кода программы lab8-1.asm

Создали исполняемый файл и запустили его. В результате получили соответствующее число проходов (рис. [7](#fig:007)).

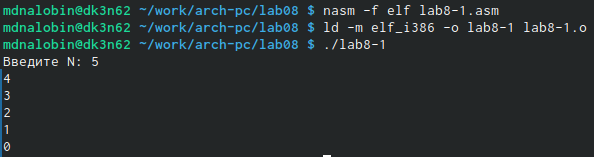


Figure 7: Запуск повторно измененной программы lab8-1

Создали файл lab8-2.asm и заполнили его текстом из Листинга 8.2. после внимательного изучения (рис. [8](#fig:008) и рис. [9](#fig:009)).

Figure 8: Создание файла lab8-2.asm

Figure 8: Создание файла lab8-2.asm

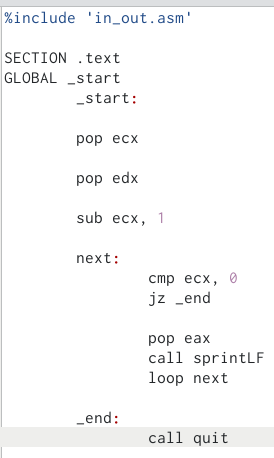


Figure 9: Код программы lab8-2.asm

Далее создали исполняемый файл и запустили его, указав аргументы. В итоге программой было выведино все три аргумента, следовательно, три аргумента было обработано (рис. [10](#fig:010)).

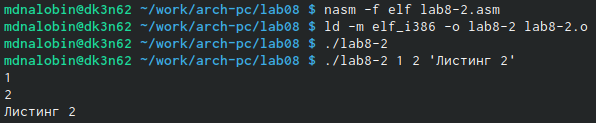


Figure 10: Запуск программы lab8-2

Создали файл lab8-3.asm и ввели в него текст Листинга 8.3. (рис. [11](#fig:011) и рис. [12](#fig:012)).

Figure 11: Создание файла lab8-3.lst

Figure 11: Создание файла lab8-3.lst

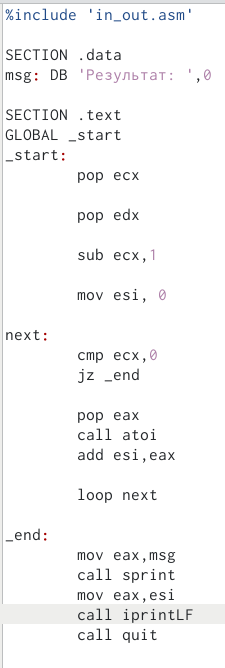


Figure 12: Код программы lab8-3.asm

Создали исполняемый файл и запустили его, указав приведенные аргументы для проверки (рис. [13](#fig:013)).

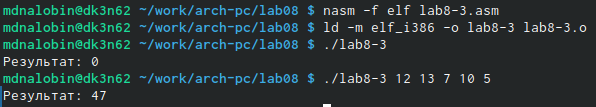


Figure 13: Запуск программы lab8-3

На этом этапе редактируем текст программы таким образом, чтобы она выводила произведение аргументов, создаем исполняемый файл и запускаем его, проверяя на работу при аргументах и их отсутствии. (рис. [14](#fig:014) и рис. [15](#fig:015)).

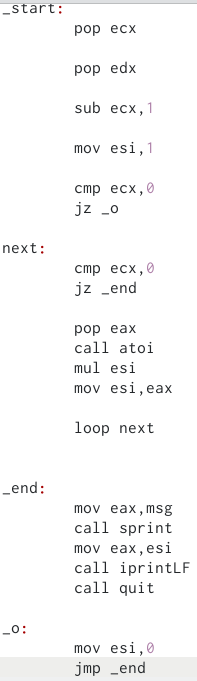


Figure 14: Код измененной программы lab8-3.asm

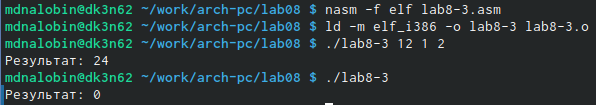


Figure 15: Запуск измененной программы lab8-3

##Самостоятельная работа

Создали файл sr.asm и написали в нем программу, которая находит сумму значений для некоторой функции, в моем случае - 4x-3, при заданных аргументах. (рис. [16](#fig:016) и рис. [17](#fig:017)).

Figure 16: Создание файла sr.asm

Figure 16: Создание файла sr.asm

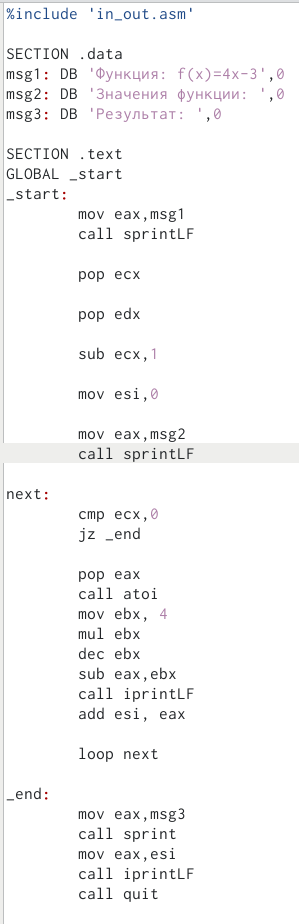


Figure 17: Код программы sr.asm

После создали исполняемый файл sr и проверили на корректность, используя два набора аргументов (рис. [18](#fig:018)).

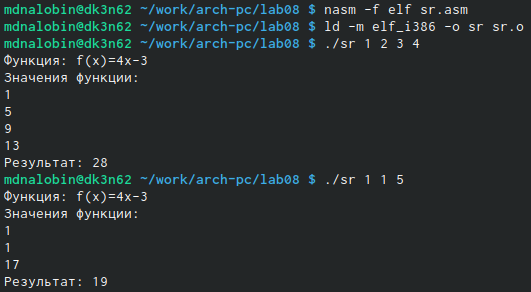


Figure 18: Запуск программы sr

# 3 Выводы

В ходе данной лабораторной работы приобрели навык использования циклов и обработки аргументов командной строки.

:::