Лабораторная работа №8

Программирование цикла. Обработка аргументов командной строки.

Турсунбоев Сардорбек

Содержание

# 1 Цель работы

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

# 2 Задание

1. Изучить операторы цикла
2. Изучить организацию стека
3. Изучить передачу аргументов
4. Рассмотреть примеры программ
5. Выполнить самостоятельное задание

# 3 Выполнение лабораторной работы

Я создал каталог для программ лабораторной работы No8 и перешел в него. Далее создал файл lab8-1.asm.

Я написал в файле lab8-1.asm текст программы из листинга 8.1. После этого создал исполняемый файл и проверил его работу.

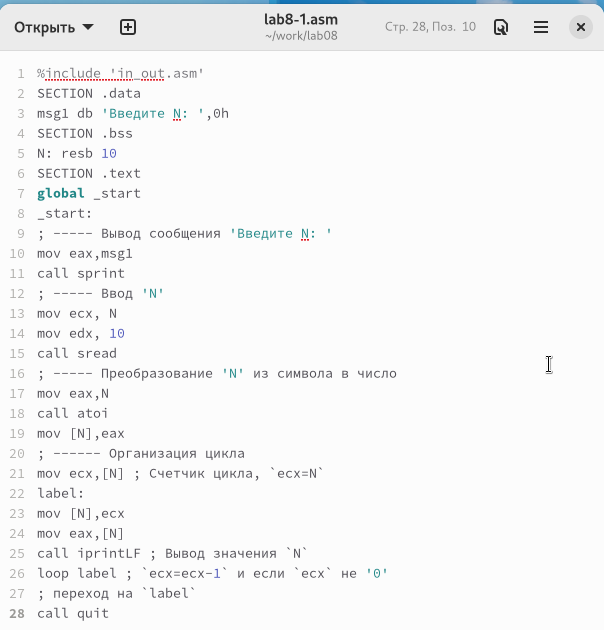


Figure 1: Код программы lab8-1.asm

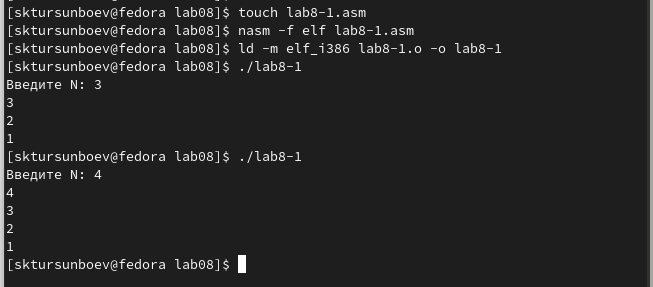


Figure 2: Тестирование программы lab8-1.asm

Этот пример показывает, что использование регистра ecx в теле цикла loop может п ривести к некорректной работе программы.

Я внес изменения в текст программы, добавив изменение значения регистра ecx в цикле. Затем создал исполняемый файл и проверил его работу. Программа запускала бесконечный цикл при нечетном значении N и выводила только нечетные числа при четном N.

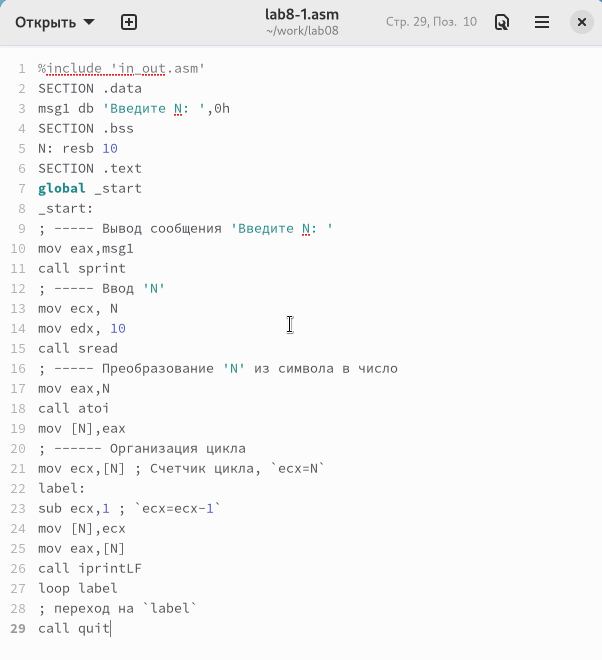


Figure 3: Код программы lab8-1.asm

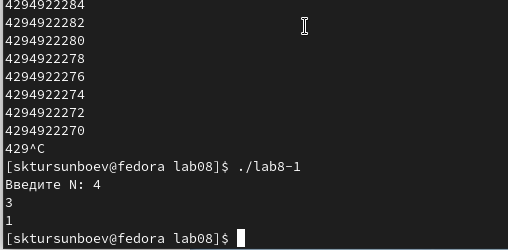


Figure 4: Тестирование программы lab8-1.asm

Для использования регистра ecx в цикле и сохранения корректности работы программы можно использовать стек. Я внес изменения в текст программы, добавив команды push и pop для сохранения значения счетчика цикла loop. После этого создал исполняемый файл и проверил его работу. Теперь программа выводила числа от N-1 до 0, и число проходов цикла соответствовало значению N.

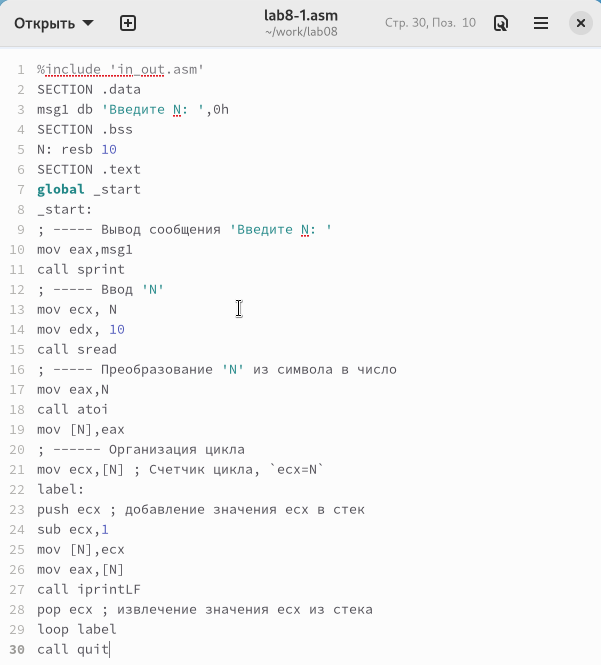


Figure 5: Код программы lab8-1.asm

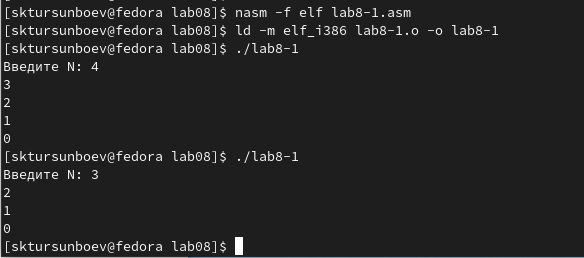


Figure 6: Тестирование программы lab8-1.asm

Я создал файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и ввел в него текст программы из листинга 8.2. Затем создал исполняемый файл, запустил его и указал 5 аргументов. Программа успешно обработала эти 5 аргументов.

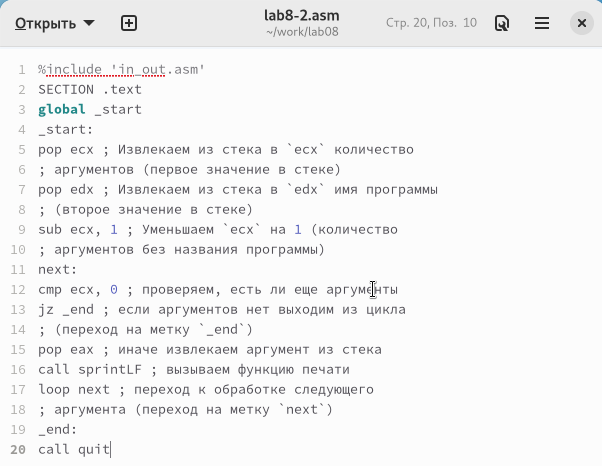


Figure 7: Код программы lab8-2.asm

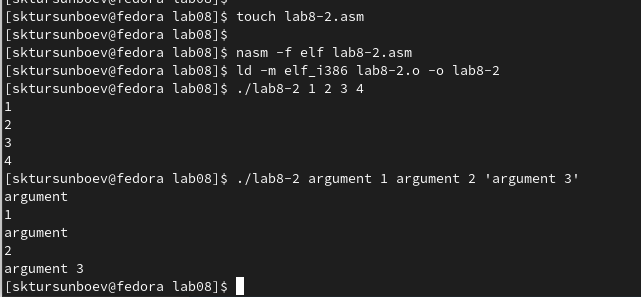


Figure 8: Тестирование программы lab8-2.asm

Рассмотрим ещё один пример программы, которая выводит сумму чисел, переданных в неё в качестве аргументов.

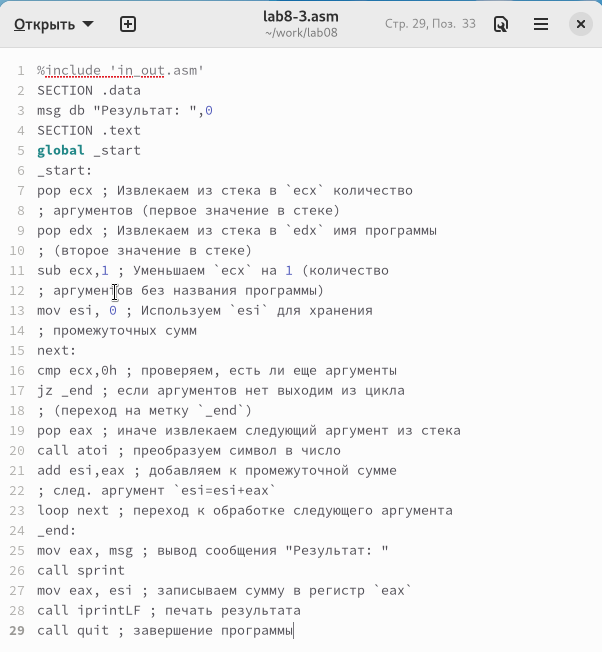


Figure 9: Код программы lab8-3.asm

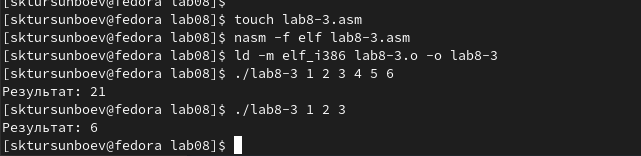


Figure 10: Тестирование программы lab8-3.asm

Я внёс изменения в текст программы из листинга 8.3, чтобы она вычисляла произведение аргументов командной строки.

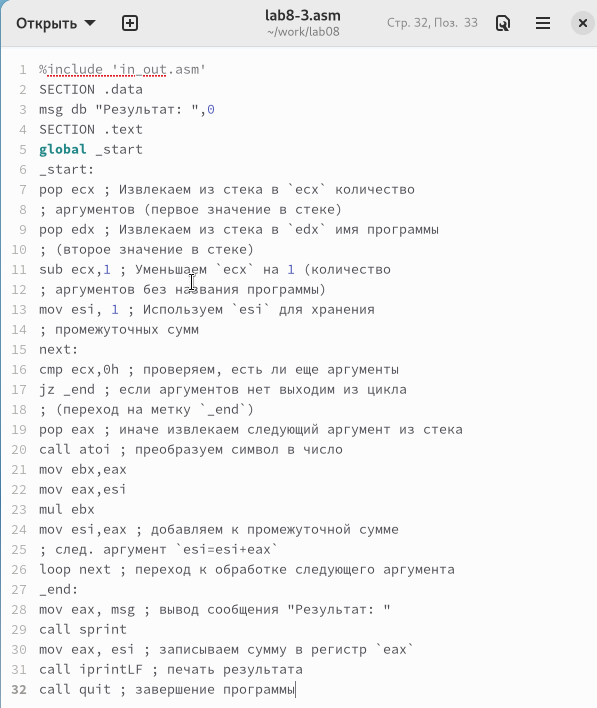


Figure 11: Код программы lab8-3.asm

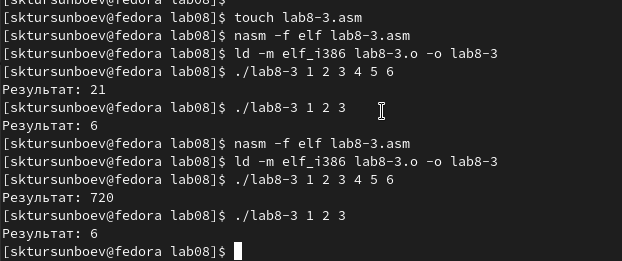


Figure 12: Тестирование программы lab8-3.asm

Написал 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 x передаются как аргументы. Вид функции f(x) выбрать из таблицы 8.1 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создал исполняемый файл и проверил его работу на нескольких наборах x.

для варианта 4

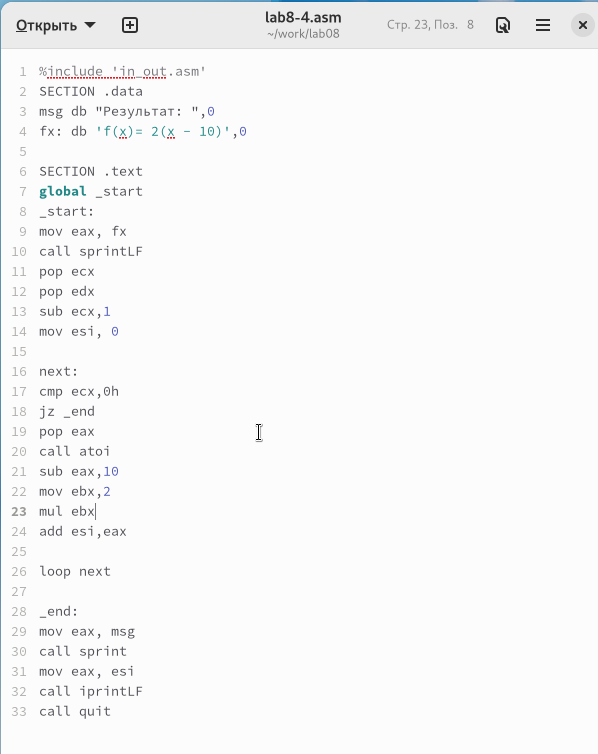


Figure 13: Код программы lab8-4.asm

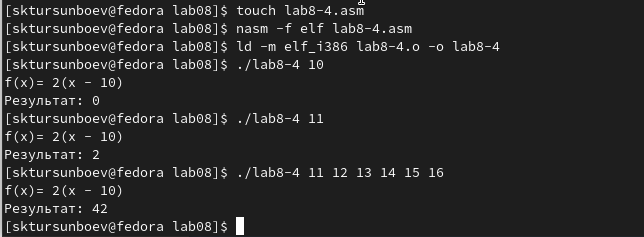


Figure 14: Тестирование программы lab8-4.asm

# 4 Выводы

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