Лабораторная работа №8

Программирование цикла. Обработка аргументов командной строки.

Самарханова Полина Тимуровна

Содержание

# 1 Цель работы

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

# 2 Задание

1. Реализация циклов в NASM
2. Обработка аргументов командной строки
3. Задание для самостоятельной работы

# 3 Теоретическое введение

Стек — это структура данных, организованная по принципу LIFO («Last In — First Out» или «последним пришёл — первым ушёл»). Стек является частью архитектуры процессора и реализован на аппаратном уровне. Для работы со стеком в процессоре есть специальные регистры (ss, bp, sp) и команды. Основной функцией стека является функция сохранения адресов возврата и передачи аргументов при вызове процедур. Кроме того, в нём выделяется память для локальных переменных и могут временно храниться значения регистров.

Стек имеет вершину, адрес последнего добавленного элемента, который хранится в регистре esp (указатель стека). Противоположный конец стека называется дном. Значение, помещённое в стек последним, извлекается первым. При помещении значения в стек указа- тель стека уменьшается, а при извлечении — увеличивается. Для стека существует две основные операции: • добавление элемента в вершину стека (push); • извлечение элемента из вершины стека (pop).

# 4 Выполнение лабораторной работы

## 4.1 Реализация циклов в NASM

Я создала директорию,в которой далее выполняла лабораторную работу (рис. ??).

Создание каталога

Создание каталога

Перехожу в созданный каталог и создаю файл lab8.asm (рис. ??).

Создание файла

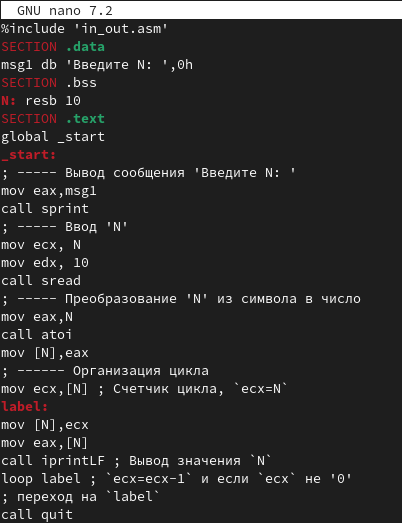
Создание файла

Далее я скопировала файл in\_out.asm (рис. ??).

Копирование файла in_out.asm

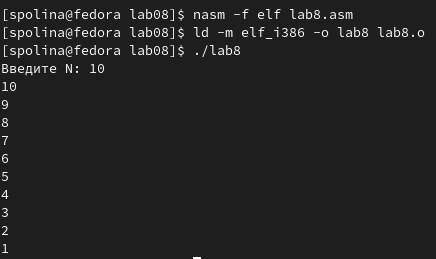
Копирование файла in\_out.asm

После чего я записала текст кода из листинга 8.1 (рис. ??). Эта программа запрашивает число N, и выдает все числа перед N вместе с ним до 0.



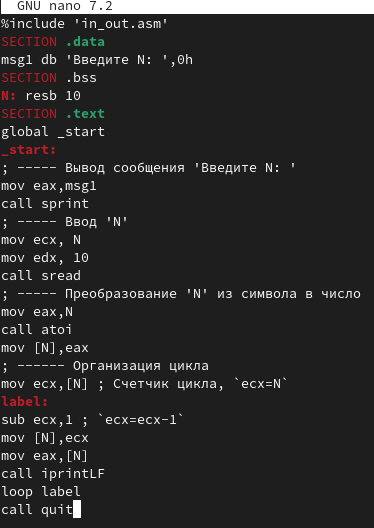
Редактирование программы

Далее я создала исполняемый код и проверила, что он работает(рис. ??).



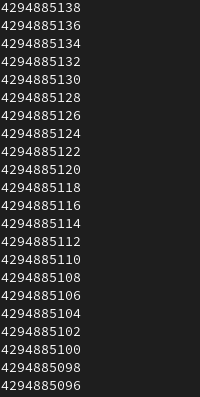
Запуск программы

После я отредактировала код,добавив изменение значение регистра ecx в цикле (рис. ??).



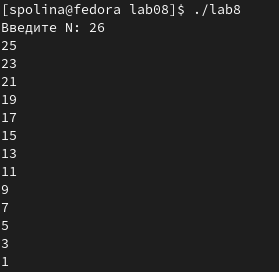
Редактирование программы

Я заапустила программу. Код стал зацикливаться и бесконечно передавать последовательные значения (рис. ??).



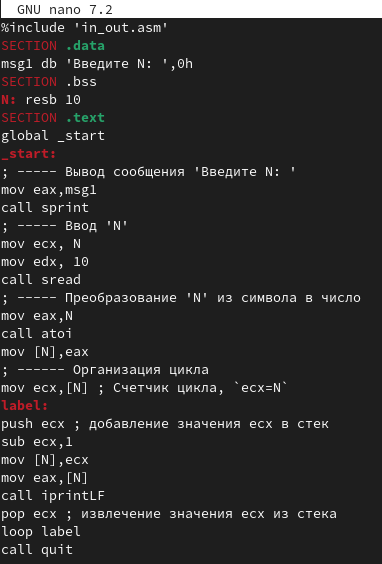
Запуск программы

Пробую еще раз запустить программу,но с другим числом, теперь она выдаёт предыдущие числа,но перескаивает через 1 (рис. ??).



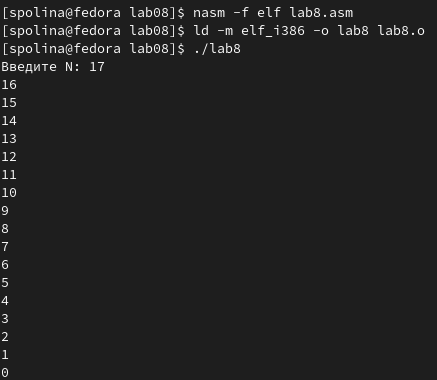
Запуск программы

Еще раз отредактировала код программы,добавив команды push и pop для сохранения значения счетчика цикла loop (рис. ??).



Редактирование программы

Создаю и запускаю исполняемый файл (рис. ??). Теперь программа показывает все предыдущие числа до 0,не включая заданное N



Запуск программы

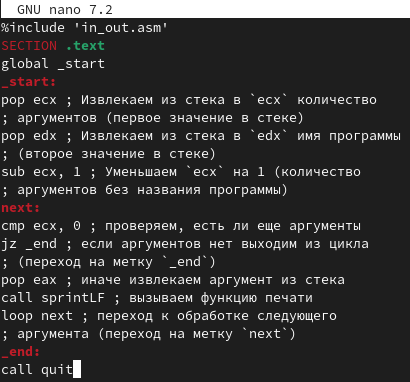
## 4.2 Обработка аргументов командной строки

Далее я создала новый файл lab8-2.asm, используя команду touch (рис. ??).

Создание файла

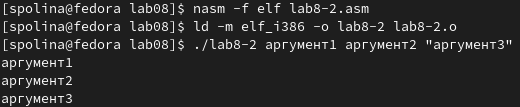
Создание файла

После я открыла файл в GNU nano и записала код из листинга 8.2 (рис. ??). Данная программа позволяет выводить на экран аргументы командной строки.



Редактирование программы

Запускаю исполняемый файл вместе с аргументами (аргумент1, аргумент2, ‘аргумент3’) (рис. ??).



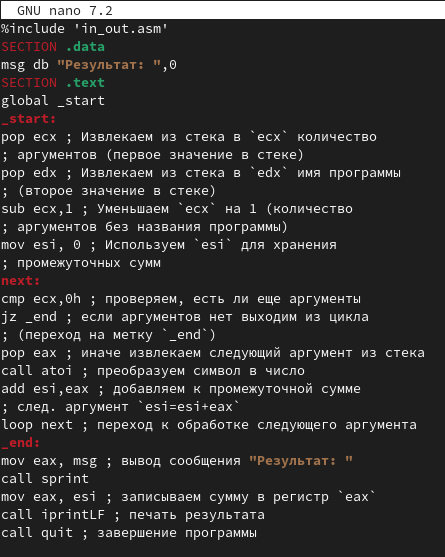
Запуск программы

Далее я создала новый файл lab8-3.asm, используя команду touch (рис. ??).

Создание файла

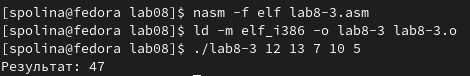
Создание файла

После я открыла файл в GNU nano и записала код из листинга 8.3 (рис. ??). Данная программа позволяет выводить на экран сумму аргументов командной строки.



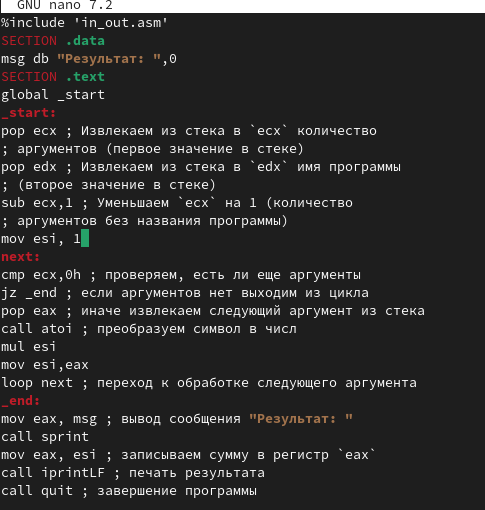
Редактирование программы

Далее я запустила исполняемый файл вместе с аргументами (рис. ??).



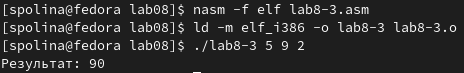
Запуск программы

Отредактировала код программы так,чтобы она выводила произведение всех аргументов (рис. ??).



Редактирование программы

После я запустила исполняемый файл вместе с аргументами (рис. ??). Программа выдаёт произведение всех аргументов.



Запуск программы

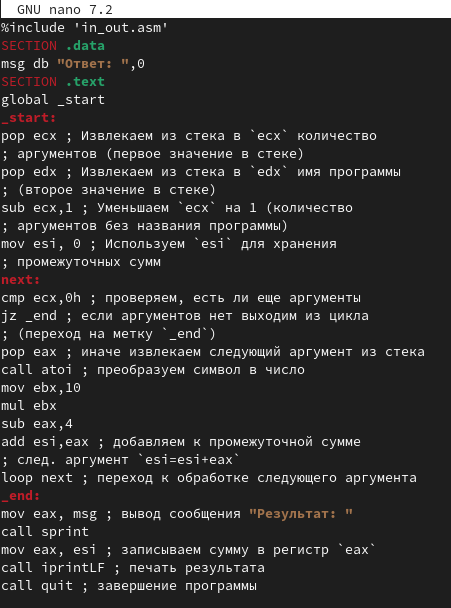
## 4.3 Задание для самостоятельной работы

Я создала файл lab8-4.asm в котором написала код для последней задачи (рис. ??).

Создание файла

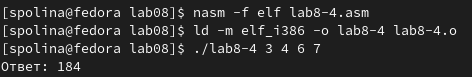
Создание файла

Далее написала код программы,который позволяет вывести сумму всех преобразованных аргументов из варианта №9(рис. ??).



Редактирование программы

Запустила исполняемый файл (рис. ??). Программа выдала верную сумму всех преобразованных аргументов.



Запуск программы

# 5 Выводы

В данной лабораторной работать я научилась работать с циклами, выводом аргументов и функций.

# Список литературы

1. Лабораторная работа №8