Отчёт по лабораторной работе №8

Дисциплина: Архитектура компьютера

Пономарева Татьяна Александровна

Содержание

# 1 Цель работы

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

# 2 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.

# 3 Выполнение лабораторной работы

## 3.1 Реализация циклов в NASM

Создаю каталог для программ лабораторной работы № 8, перехожу в него и создаю файл lab8-1.asm (рис. 1).

Терминал. Создание каталога lab08. Создание файла lab8-1.asm

Рис. 1: Терминал. Создание каталога lab08. Создание файла lab8-1.asm

Ввожу в файл lab8-1.asm текст программы из листинга 8.1 (рис. 2).

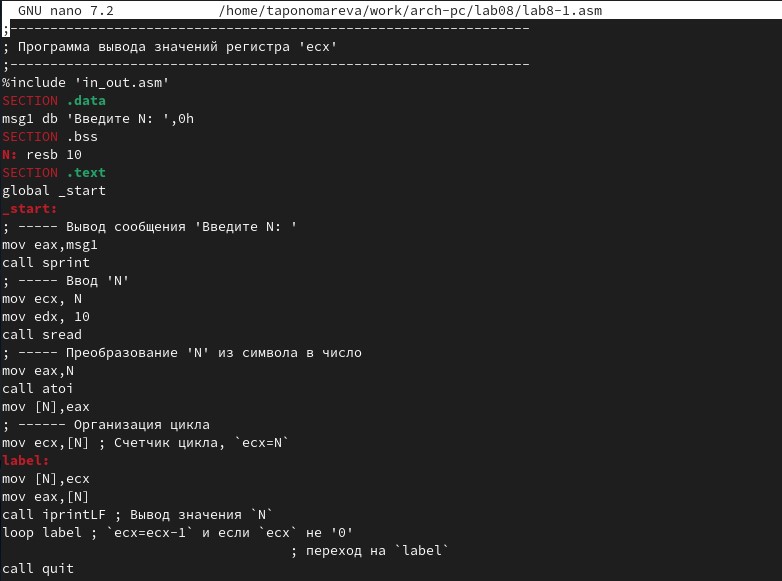


Рис. 2: Окно Midnight Commander. Содержание файла lab8-1.asm

Создаю исполняемый файл и проверяю его работу (рис. 3). Программа демонстрирует работу циклов в NASM.

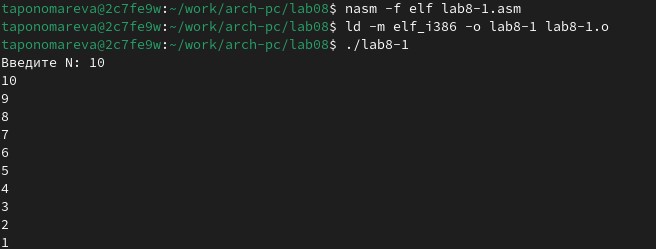


Рис. 3: Терминал. Создание исполняемого файла lab8-1. Проверка работы lab8-1

Изменяю текст программы, добавив изменение значения регистра ecx в цикле (рис. 4).

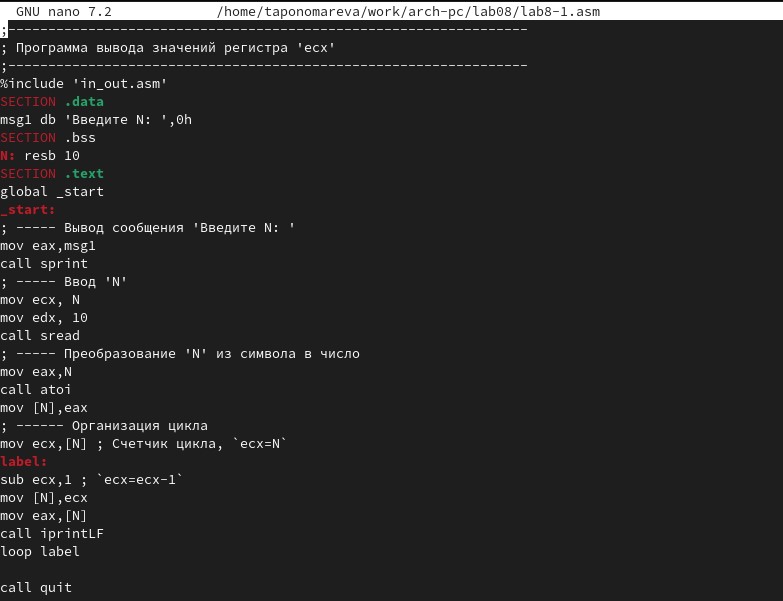


Рис. 4: Окно Midnight Commander. Содержание измененного файла lab8-1.asm

Создаю исполняемый файл с измененным содержанием и проверяю его работу (рис. 5). Поскольку значение регистра ecx уменьшается на 2 в каждой итерации, общее количество итераций сокращается вдвое. Число проходов цикла не соответствует значению N, введенному с клавиатуры.

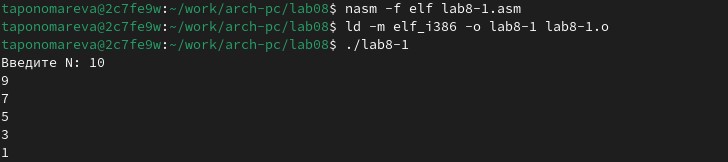


Рис. 5: Терминал. Создание исполняемого файла lab8-1. Проверка работы lab8-1

Внесу изменения в текст программы, добавив команды push и pop для сохранения значения счетчика цикла loop (рис. 6).

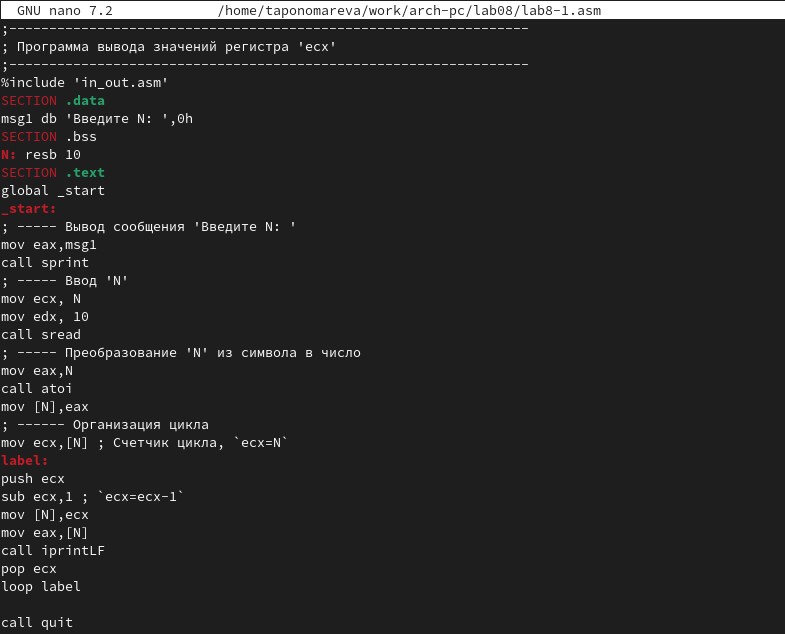


Рис. 6: Окно Midnight Commander. Содержание измененного файла lab8-1.asm

Снова создаю исполняемый файл и проверяю его работу (рис. 7). Заметим, что число проходов цикла соответствует значению N, введеному с клавиатуры, но выводимые значения смещены на -1.

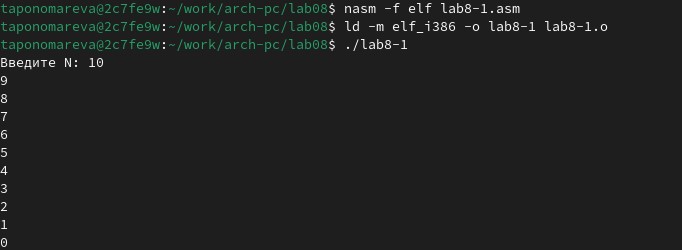


Рис. 7: Терминал. Создание исполняемого файла lab8-1. Проверка работы lab8-1

## 3.2 Обработка аргументов командной строки

Создаю файл lab8-2.asm в каталоге ~/work/arch-pc/lab08 и ввожу в него текст программы из листинга 8.2 (рис. 8).

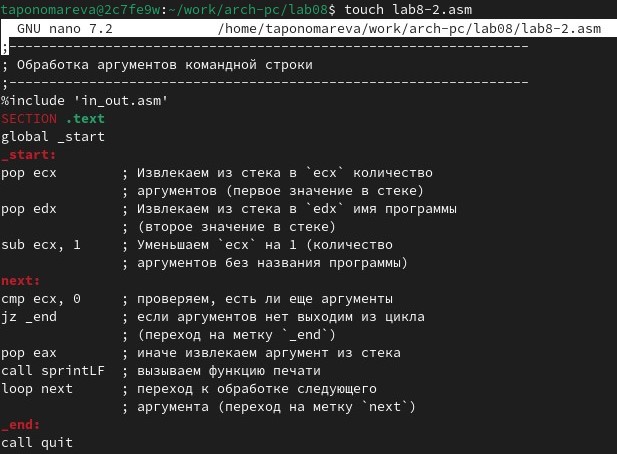


Рис. 8: Создание файла lab8-2.asm. Содержание файла lab8-2.asm

Компилирую исполняемый файл и проверяю его выполнение (рис. 9). Программой было обработано 3 аргумента.

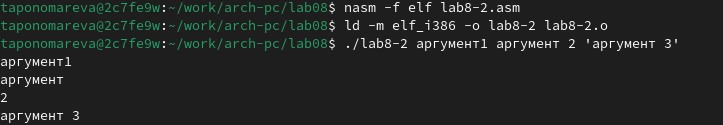


Рис. 9: Терминал. Компиляция исполняемого файла lab8-2. Проверка работы lab8-2

Создаю файл lab8-3.asm в каталоге ~/work/archpc/lab08 и ввожу в него текст программы из листинга 8.3 (рис. 10).

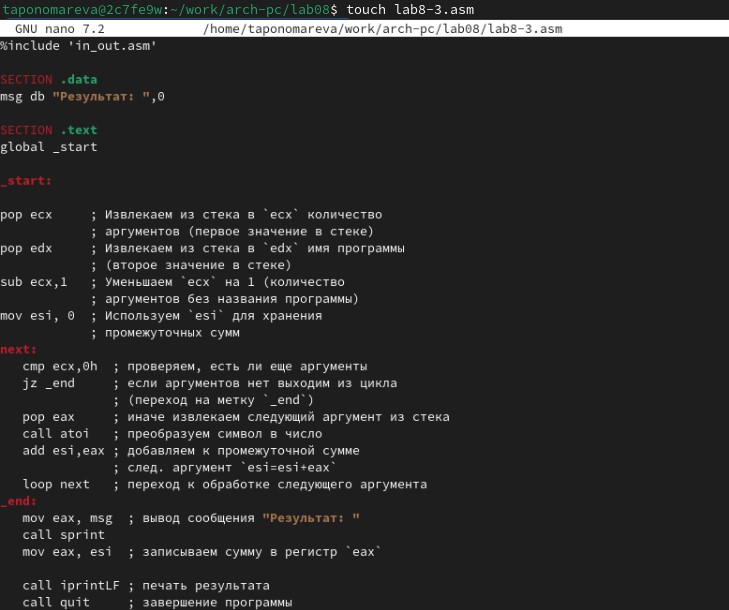


Рис. 10: Создание файла lab8-3.asm. Содержание файла lab8-3.asm

Создаю исполняемый файл и запускаю его, указав аргументы (рис. 11). Результат программы равен 47.

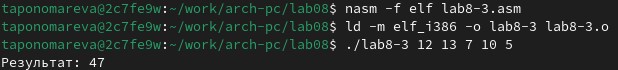


Рис. 11: Терминал. Создание исполняемого файла lab8-3. Проверка работы lab8-3

Изменяю текст программы из листинга 8.3 для вычисления произведения аргументов командной строки (рис. 12).

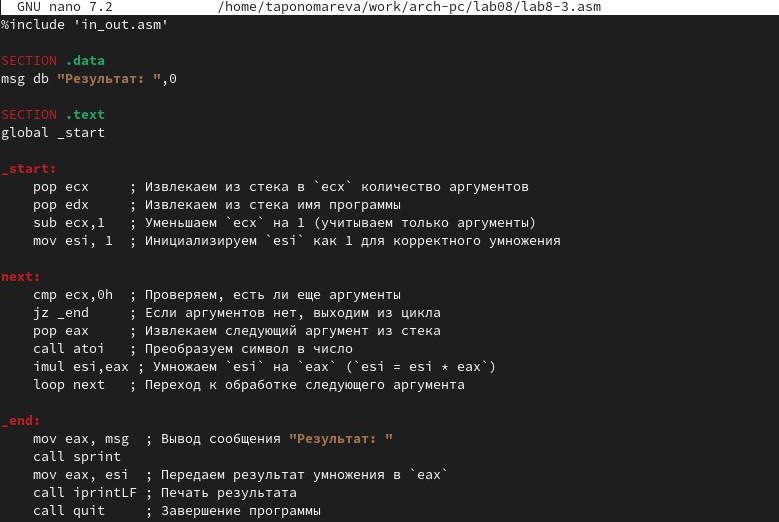


Рис. 12: Окно Midnight Commander. Содержание измененного файла lab8-3.asm

Создаю исполняемый файл и запускаю его (рис. 13). Программа перемножает вводимые числа.

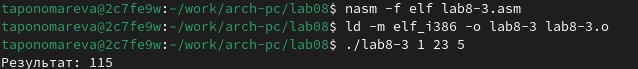


Рис. 13: Терминал. Создание исполняемого файла lab8-3. Проверка работы lab8-3

# 4 Задание для самостоятельной работы

У меня вариант №3.

Пишу программу, которая находит сумму значений функции f(x)=10x-5 для x = x1,x2,x3,…,xn, т.е. выводит f(x1)+f(x2)+…+f(xn).

Код программы для варианта №3

%include 'in\_out.asm'  
  
SECTION .data  
msg\_function db "Функция: f(x) = 10x - 5", 0  
msg\_res db "Результат: ", 0  
  
SECTION .text  
GLOBAL \_start  
  
\_start:  
 mov eax, msg\_function  
 call sprintLF  
  
 pop ecx  
 pop edx  
 sub ecx, 1  
 mov esi, 0  
  
next:  
 cmp ecx, 0h  
 jz \_end  
 pop eax  
 call atoi  
  
 mov ebx, 10  
 mul ebx  
 sub eax, 5  
 add esi, eax  
 loop next  
  
\_end:   
 mov eax, msg\_res  
 call sprint  
 mov eax, esi  
 call iprintLF  
 call quit

Создаю исполняемый файл lab8-4 и запускаю его (рис. 14). Программа работает корректно (рис. 14).

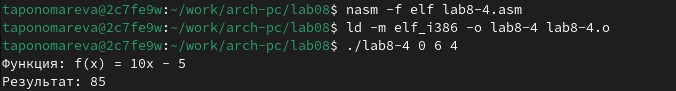


Рис. 14: Терминал. Создание исполняемого файла lab8-4. Проверка работы lab8-4

Загружаю отчёт на GitHub.

# 5 Выводы

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

# Список литературы

1. [Курс на ТУИС](https://esystem.rudn.ru/course/view.php?id=112)
2. [Лабораторная работа №8](https://esystem.rudn.ru/pluginfile.php/2089095/mod_resource/content/0/%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%E2%84%968.%20%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%20%D1%86%D0%B8%D0%BA%D0%BB%D0%B0.%20%D0%9E%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0%20%D0%B0%D1%80%D0%B3%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%BE%D0%B2%20%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D0%BD%D0%BE%D0%B9%20%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%B8..pdf)