Отчет по лабораторной работе №8

Дисциплина: архитектура компьютера

Луцкая Алиса Витальевна

Содержание

# 1 Цель работы

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

# 2 Задание

1. Реализация циклом в NASM
2. Обработка аргументов командной строки
3. Задания 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.

# 4 Выполнение лабораторной работы

## 4.1 Реализация циклов в NASM

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

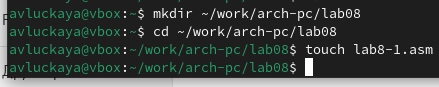


Рис. 1: Создание каталога и файла, перемещение между деррикториями

Ввожу в созданный файл программу из листинга. (рис. -fig. 2).

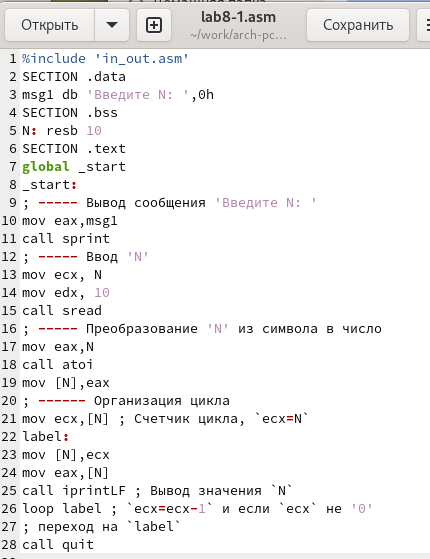


Рис. 2: Ввод программы из листинга

Создаю исполняемый файл и проверяю его работу (рис. -fig. 3).

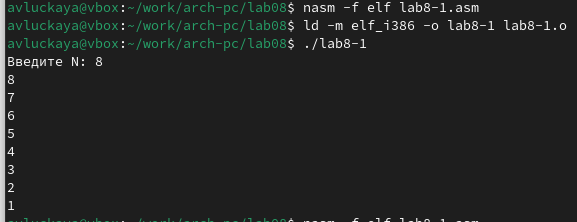


Рис. 3: Запуск программы

Заменяю программу изначальную так, что в теле цикла я изменяю значение регистра ecx (рис. -fig. 4).

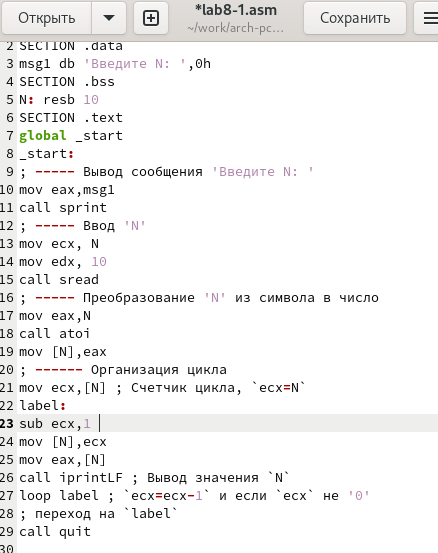


Рис. 4: Изменение программы

Из-за того, что теперь регистр ecx на каждой итерации уменьшается на 2 значения, количество итераций уменьшается вдвое (рис. -fig. 5).

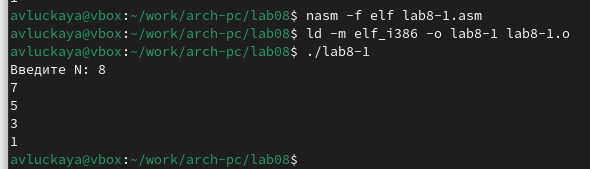


Рис. 5: Запуск измененной программы

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

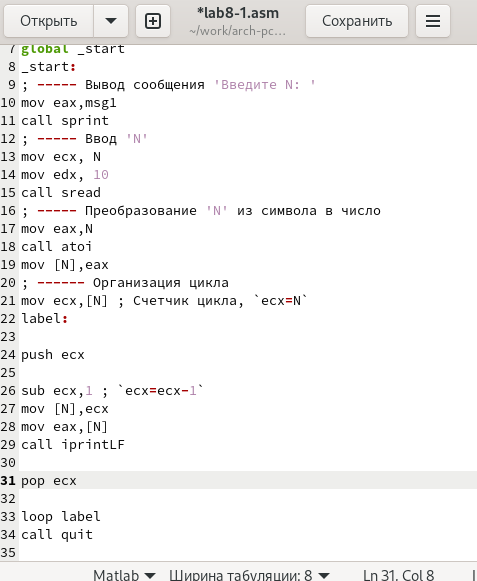


Рис. 6: Добавление push и pop в цикл программы

Количество итераций совпадает с введенным N, но произошло смещение выводимых чисел на -1 (рис. -fig. 7).

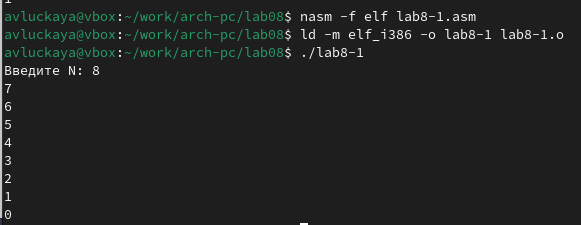


Рис. 7: Запуск измененной программы

## 4.2 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. -fig. 8).

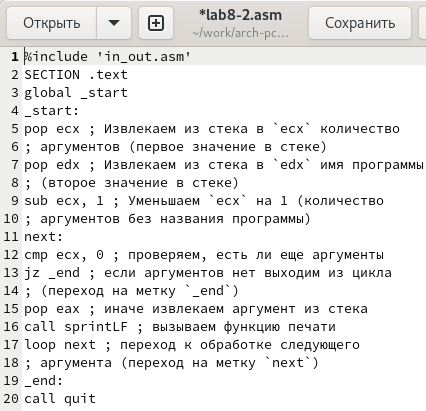


Рис. 8: Ввод программы из листинга

Создаю исполняемый файл и запускаю его, указав аргументы: аргумент1 аргумент 2 ‘аргумент 3’ (рис. -fig. 9).

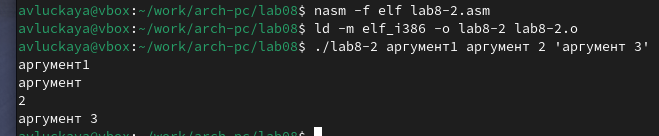


Рис. 9: Запуск второй программы

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

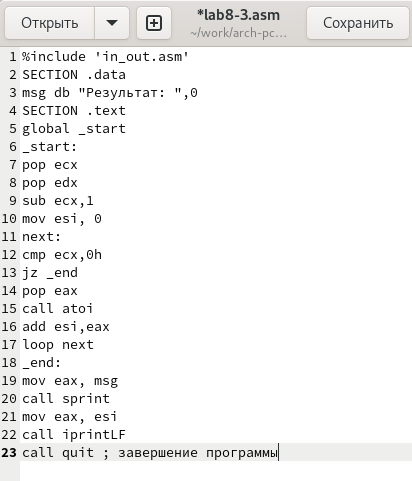


Рис. 10: Ввод программы из третьего листинга

Создаю исполняемый файл и запускаю его, указав аргументы 12 13 7 10 5.(рис. -fig. 11).

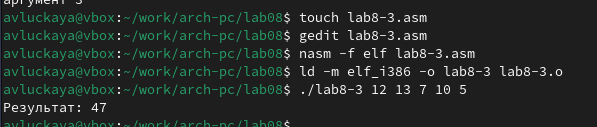


Рис. 11: Запуск третьей программы

Изменяю программу так, чтобы указанные аргументы она умножала, а не складывала (рис. -fig. 12).

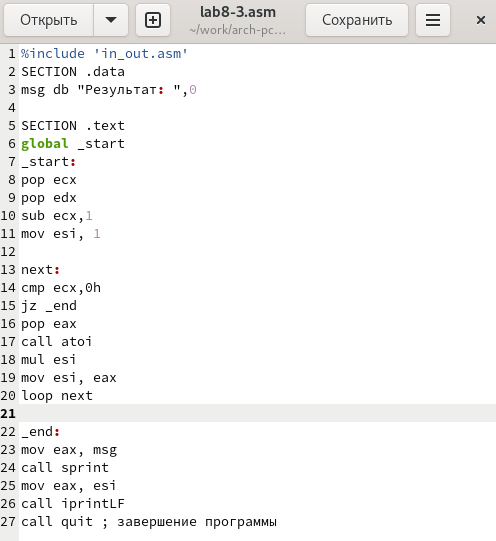


Рис. 12: Изменение третьей программы

Создаю исполняемый файл и запускаю его. (рис. -fig. 13). Программа действительно перемножает вводимые аргументы.

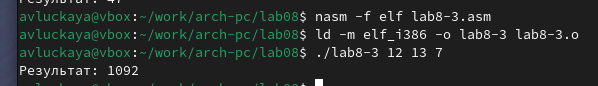


Рис. 13: Запуск измененной третьей программы

## 4.3 Задание для самостоятельной работы

Пишу программму, которая будет находить сумма значений для функции f(x) = 15x+2, которая совпадает с моим 11 варинтом (рис. -fig. 14).

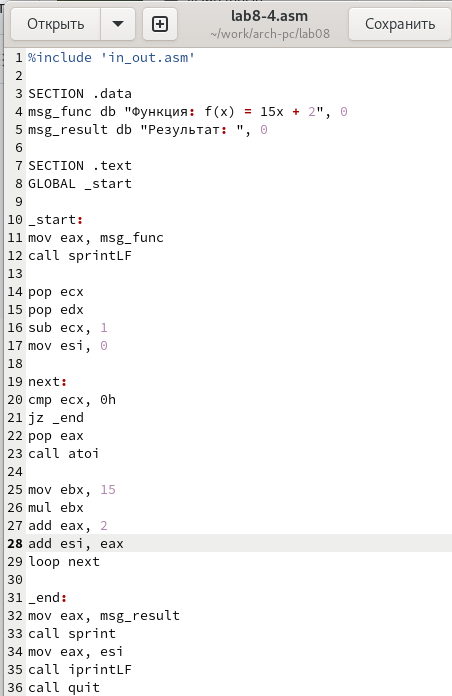


Рис. 14: Написание программы для самостоятельной работы

Создаю исполняемый файл и запускаю его. Проверяю работу программы с разными аргументами (рис. -fig. 15). Программа работает корректно.

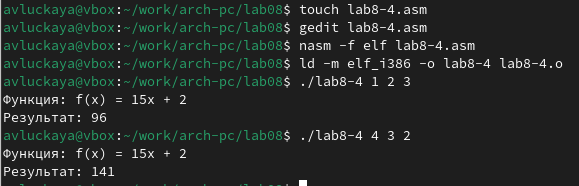


Рис. 15: Запуск измененной третьей программы

Код программы:

%include 'in\_out.asm'  
  
SECTION .data  
msg\_func db "Функция: f(x) = 15x + 2", 0   
msg\_result db "Результат: ", 0  
  
SECTION .text  
GLOBAL \_start  
  
\_start:  
mov eax, msg\_func  
call sprintLF  
  
pop ecx   
pop edx   
sub ecx, 1   
mov esi, 0   
  
next:  
cmp ecx, 0h  
jz \_end  
pop eax   
call atoi  
  
mov ebx, 15   
mul ebx   
add eax, 2   
add esi, eax   
loop next  
  
\_end:  
mov eax, msg\_result  
call sprint  
mov eax, esi  
call iprintLF  
call quit

# 5 Выводы

В результате выполнения данной лабораторной работы я приобрел навыки написания программ с использованием циклов а также научился обрабатывать аргументы командной строки.

# Список литературы

1. https://esystem.rudn.ru/course/view.php?id=112