Отчёт по лабораторной работе №8

Дисциплина:архитектура компьютера

Лобанов Владислав Олегович

Содержание

# Цель работы

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

# Задание

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

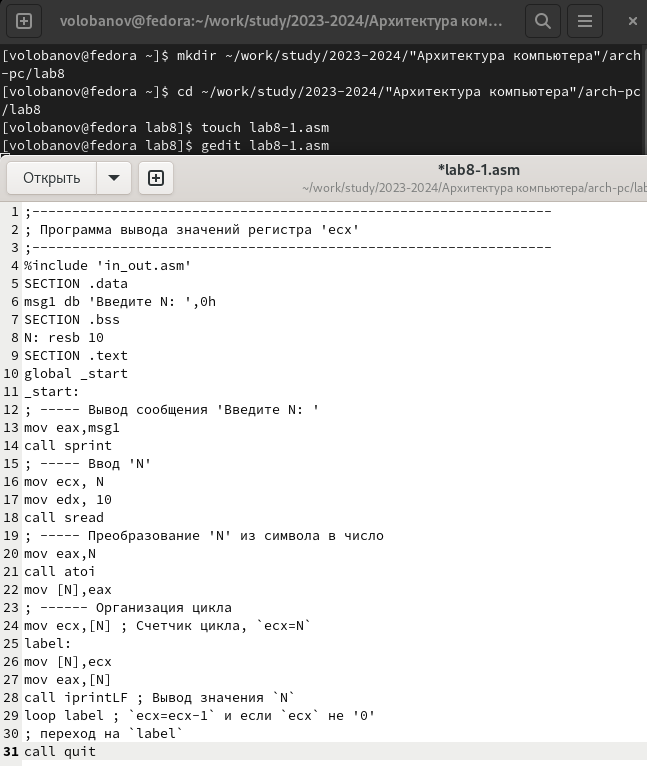
# Теоретическое введение

Цикл в программировании — это управляющая конструкция, которая заставляет какой-то блок кода выполняться несколько раз.

# Выполнение лабораторной работы

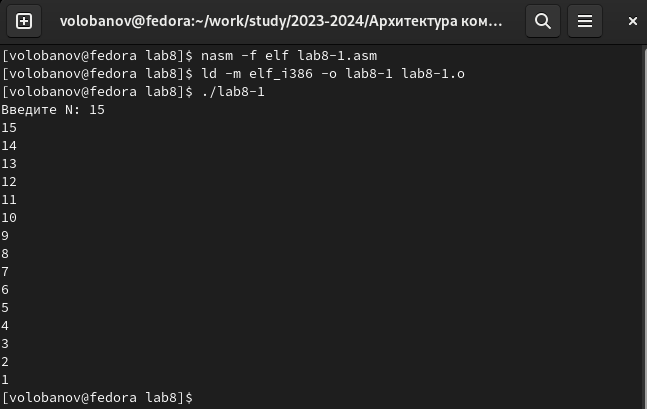
## Реализация циклов в NASM

Создаю каталог для программ для лабораторной работы №8, перехожу в него и создаю файл lab8-1.asm, в который ввожу программу вывода значений регистра ecx (рис. @fig:001).



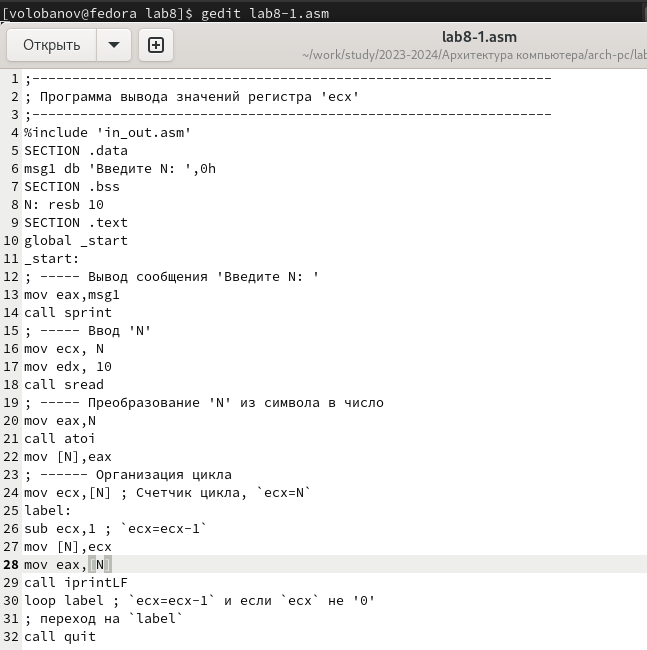
Создание каталога и файла в ней, редактирование файла

Создаю исполняемый файл и запускаю его (рис. @fig:002). Программа отработала корректно.



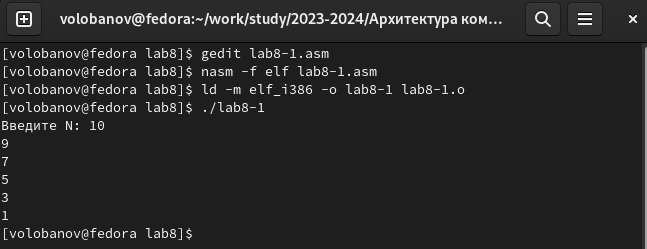
Исполнение программы

Изменяю текст программы добавив изменение значения регистра ecx в цикле (рис. @fig:003).



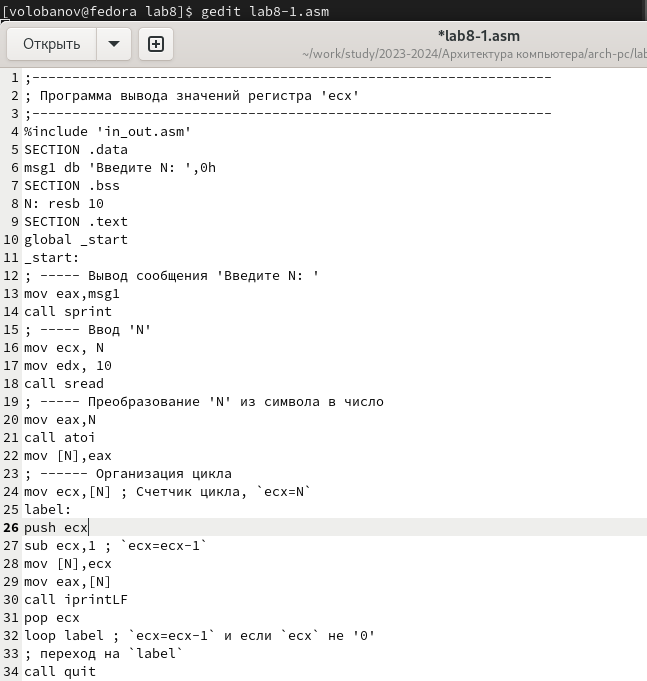
Изменение текста программы

Создаю исполняемый файл и запускаю его (рис. @fig:004). Число проходов цикла равно N/2.



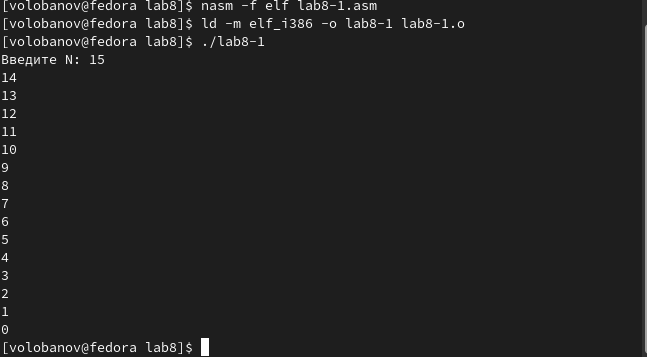
Исполнение программы

Вношу изменения в текст программы добавив команды push и pop для сохранения значения счётчика loop(рис. @fig:005).



Изменение программы

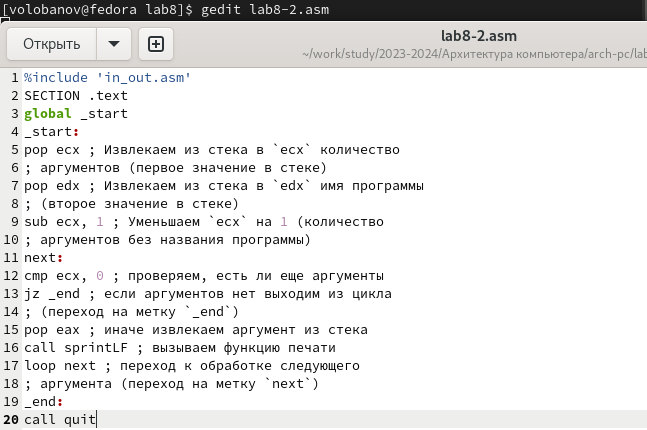
Создаю исполняемый файл и проверяю его работу (рис. @fig:006). Число проходов соответствует значению N введённому с клавиатуры.



Исполнение программы

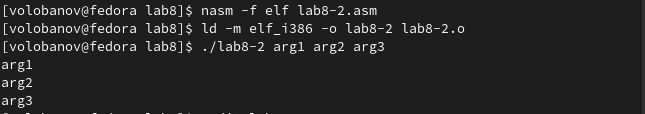
## Обработка аргументов командной строки

Создаю файл lab8-2.asm и ввожу в него программу обработки аргументов командной строки (рис. @fig:007).



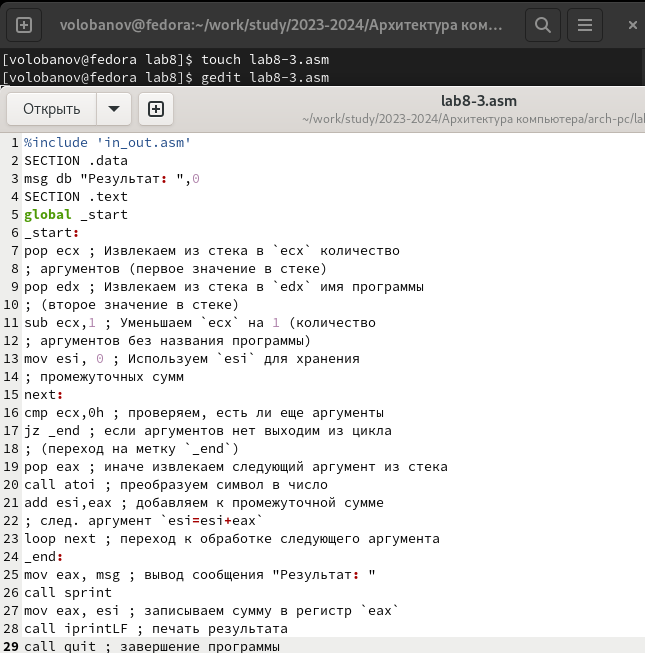
Создание и редактирование файла

Создаю исполняемый файл. Запускаю исполняемый файл (рис. @fig:008). Программой были обработаны все аргументы.



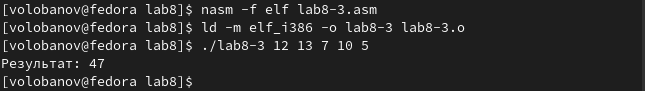
Запуск исполняемого файла

Создаю файл lab8-3.asm и ввожу в него программу вычисления суммы аргументов командной строки (рис. @fig:009).



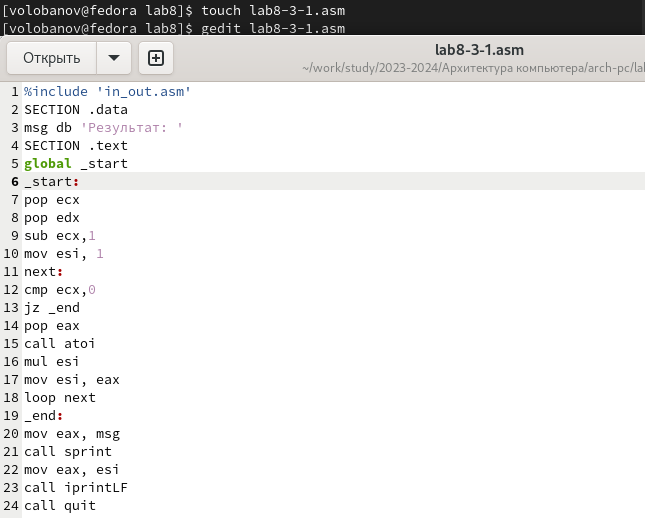
Создание файла и его редактирование

Создаю исполняемый файл и запускаю его, указав аргументы 12, 13, 7, 10, 5 (рис. @fig:010). Программа отработала корректно.



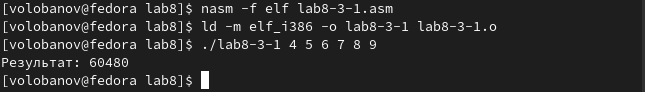
Запуск исполняемого файла

Создаю файл lab8-3-1.asm и ввожу в него программу вычисления произведения аргументов командной строки (рис. @fig:011).



Создание файла и его редактирование

Создаю исполняемый файл и запускаю его, указав аргументы 4 5 6 7 8 9 (рис. @fig:012). Программа отработала корректно.



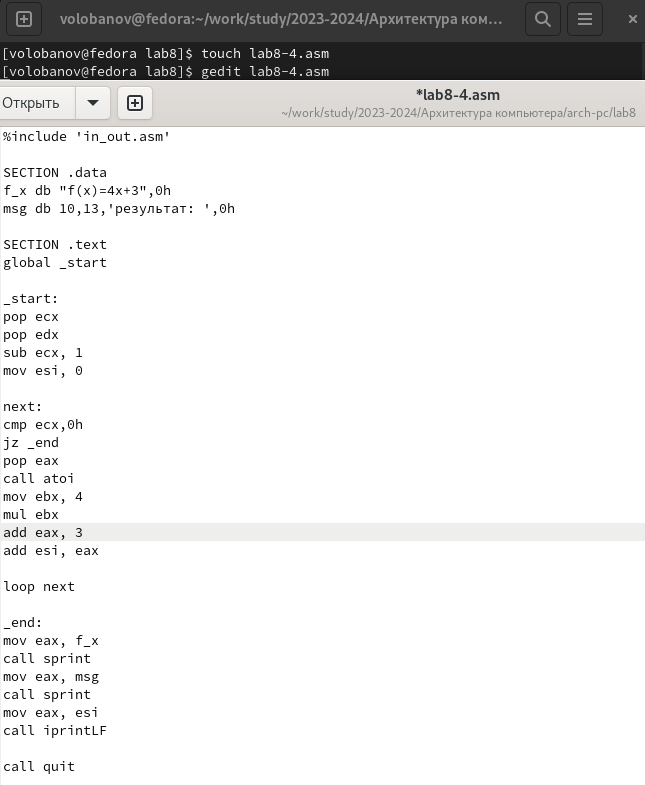
Запуск исполняемого файла

Текст программы из файла lab8-3-1.asm:

%include "in\_out.asm"  
SECTION .data  
msg db 'результат: '  
SECTION .text  
GLOBAL \_start  
  
\_start:  
pop ecx  
pop edx  
sub ecx,1  
mov esi,1  
  
next:  
cmp ecx,0  
jz \_end  
  
pop eax  
call atoi  
mul esi  
mov esi, eax  
  
loop next  
  
\_end:  
mov eax, msg  
call sprint  
mov eax, esi  
call iprintLF  
call quit

## Задание для самостоятельной работы (Вариант 20)

Создаю файл lab8-4.asm и ввожу текст программы для вычисления суммы значений функции f(x)=3(10+x) (рис. @fig:013).



Создание и редактирование файла

Текст программы из файла lab8-3-1.asm:

%include ‘in\_out.asm’

SECTION .data f\_x db “f(x)=4x+3”,0h msg db 10,13,‘результат:’,0h

SECTION .text global \_start

\_start: pop ecx pop edx sub ecx, 1 mov esi, 0

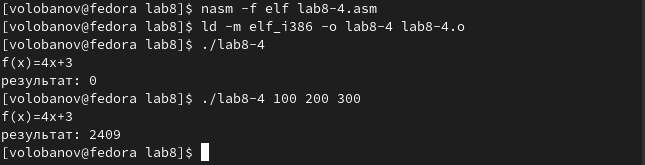
next: cmp ecx,0h jz \_end pop eax call atoi mov ebx, 4  
mul ebx add eax, 3 add esi, eax

loop next

\_end: mov eax, f\_x call sprint mov eax, msg call sprint mov eax, esi call iprintLF

call quit

Создаю исполняемый файл и запускаю исполняемый файл с разными аргументами (рис. @fig:014). Программа отработала корректно.



Запуск исполняемого файла

# Выводы

Были получены навыки по организации циклов и работе со стеком на языке NASM.

# Список литературы

1. [Лабораторная работа №8](https://esystem.rudn.ru/pluginfile.php/2089548/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)