Отчёт по лабораторной работе №6

Дисциплина: архитектура компьютера

Чернятьева Олеся Олеговна

Содержание

# 1 Цель работы

Цель данной лабораторной работы - освоение арифметческих инструкций низкоуровневого языка ассемблера NASM.

# 2 Выполнение лабораторной работы

## 2.1 Символьные и численные данные в NASM

**1**

С помощью mkdir создаю директорию lab06, перехожу в нее и создаю файл для работы. (рис. [[1](#fig:001)])

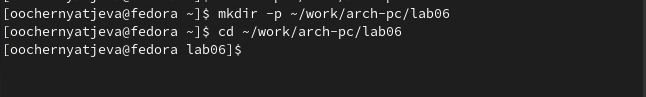


Figure 1: Создание директории

**2**

копирую файл in\_out.asm в новый созданный каталог, т.к. он будет использоваться в других программах (рис. [[2](#fig:002)]).

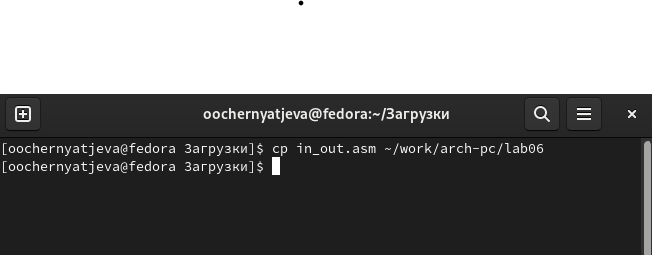


Figure 2: Создание и копирование файла для дальнейшей работы

**3**

Создаю файл lab6-1.asm и вставляю в него программу для вывода значения записанные в регистр eax (рис. [[3](#fig:003)]).

Figure 3: Редактирование файла

Figure 3: Редактирование файла

**4**

Открываю созданный файл lab6-1.asm, вставляю в него программу (рис. [[4](#fig:004)]).

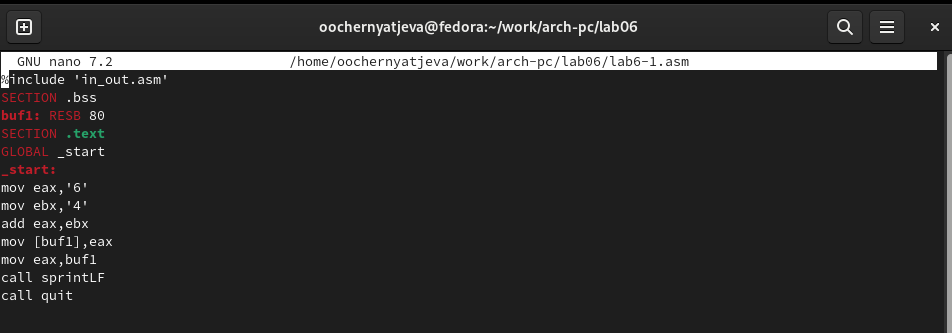


Figure 4: Редактирование файла

**5**

Создаю исполняемый файл программы и запускаю его (рис. [[5](#fig:005)]). Вывод программы: символ j, потому что программа вывела символ, соответствующий по системе ASCII сумме двоичных кодов символов 4 и 6.

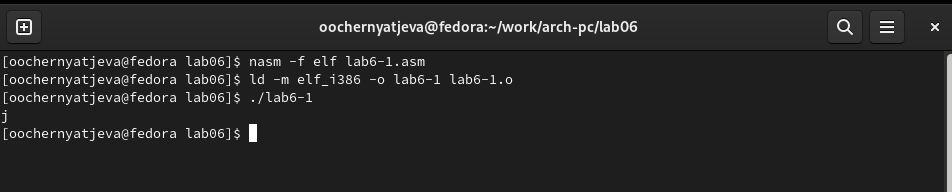


Figure 5: Запуск исполняемого файла

**6**

Изменяю в тексте программы символы “6” и “4” на цифры 6 и 4 (рис. [[6](#fig:006)]).

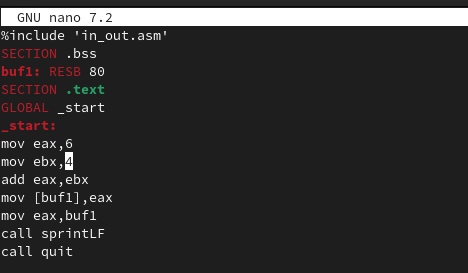


Figure 6: Редактирование файла

**7**

Создаю новый исполняемый файл программы и запускаю его (рис. [[7](#fig:007)]). Теперь вывелся символ с кодом 10, это символ перевода строки, этот символ не отображается при выводе на экран.

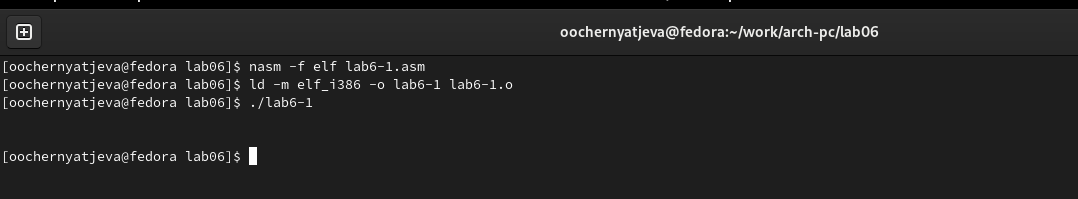


Figure 7: Запуск исполняемого файла

**8**

Создаю новый файл lab6-2.asm с помощью утилиты touch (рис. [[8](#fig:008)]).



Figure 8: Создание файла

**9**

Ввожу в файл текст другой программы для вывода значения регистра eax (рис. [[9](#fig:009)]).

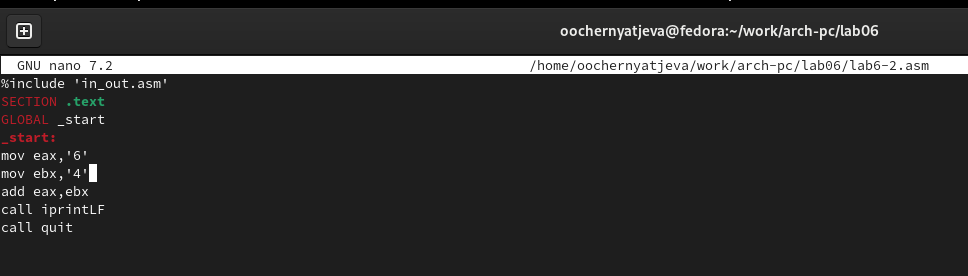


Figure 9: Редактирование файла

**10**

Создаю и запускаю исполняемый файл lab6-2 (рис. [[10](#fig:0010)]). Теперь вывод число 106, потому что программа позволяет вывести именно число, а не символ, хотя все еще происходит именно сложение кодов символов “6” и “4”.

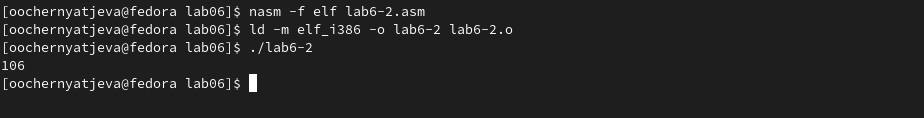


Figure 10: Запуск исполняемого файла

**11**

Заменяю в тексте программы в файле lab6-2.asm символы “6” и “4” на числа 6 и 4 (рис. [[11](#fig:011)]).



Figure 11: Редактирование файла

**12**

Создаю и запускаю новый исполняемый файл (рис. [[12](#fig:012)]).. Теперь программа складывает не соответствующие символам коды в системе ASCII, а сами числа, поэтому вывод 10.

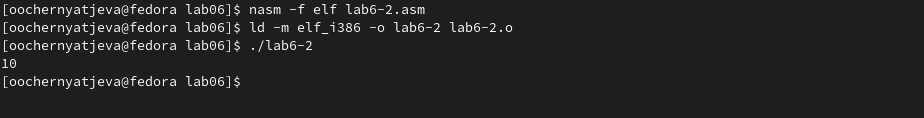


Figure 12: Запуск исполняемого файла

**13**

Заменяю в тексте программы функцию iprintLF на iprint (рис. [[13](#fig:013)]).

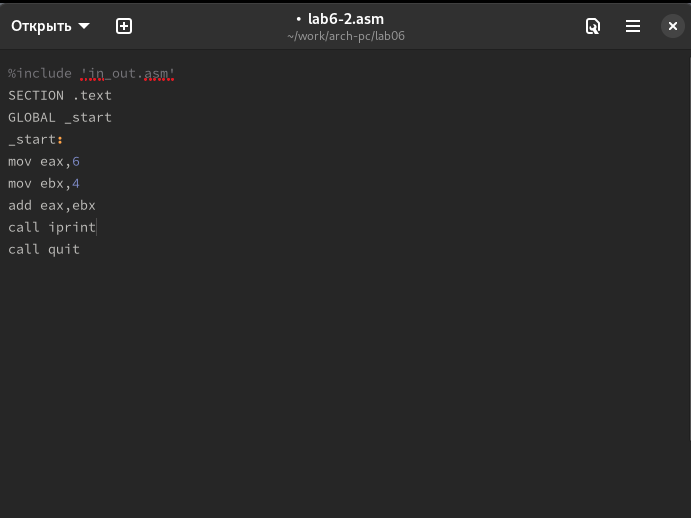


Figure 13: Редактирование файла

**14**

Создаю и запускаю новый исполняемый файл (рис. [[14](#fig:014)]). Вывод не изменился, потому что символ переноса строки не отображался, когда программа исполнялась с функцией iprintLF, а iprint не добавляет к выводу символ переноса строки, в отличие от iprintLF.

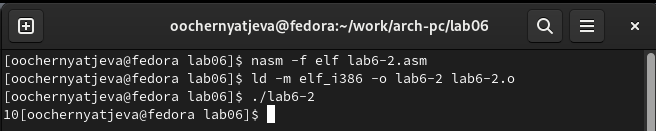


Figure 14: Запуск исполняемого файла

## 2.2 Выполнение арифметических операций в NASM

**15**

Создаю файл lab6-3.asm с помощью утилиты touch (рис. [[15](#fig:015)]).

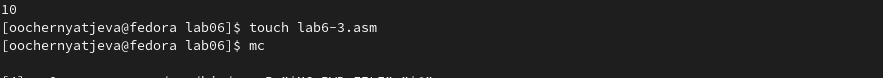


Figure 15: Создание файла

**16**

Ввожу в созданный файл текст программы для вычисления значения выражения f(x) = (5 \* 2 + 3)/3 (рис. [[16](#fig:016)]).

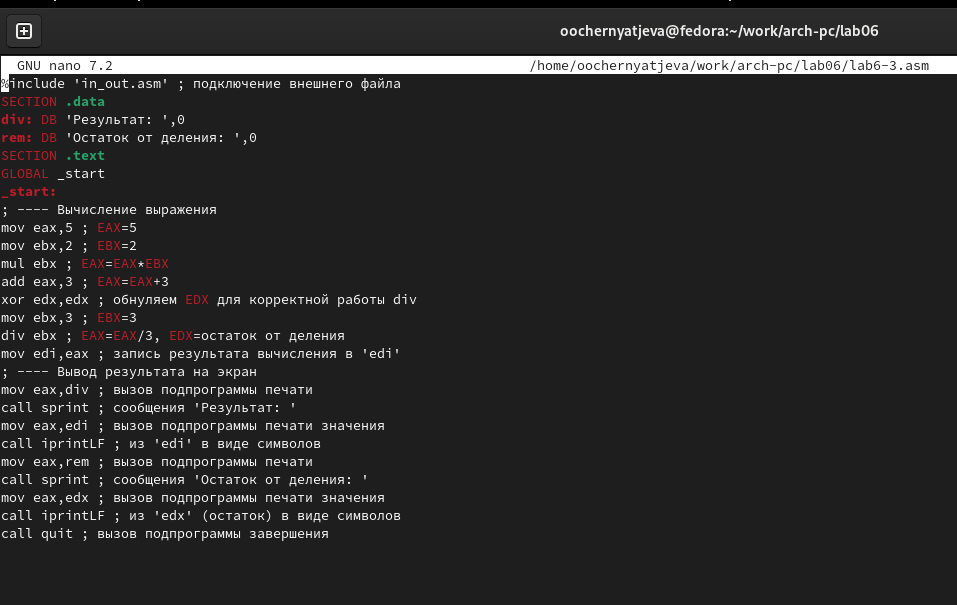


Figure 16: Редактирование файла

**17**

Создаю исполняемый файл и запускаю его (рис. [[17](#fig:017)]).

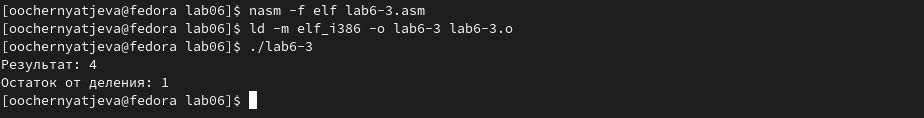


Figure 17: Запуск исполняемого файла

**18**

Изменяю программу так, чтобы она вычисляла значение выражения f(x) = (4 \* 6 + 2)/5 (рис. [[18](#fig:018)]).

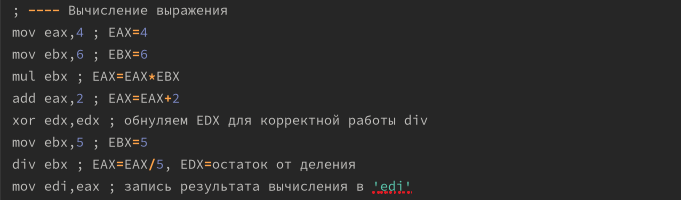


Figure 18: Изменение программы

**19**

Создаю и запускаю новый исполняемый файл (рис. [[19](#fig:019)]). Программа отработала верно.

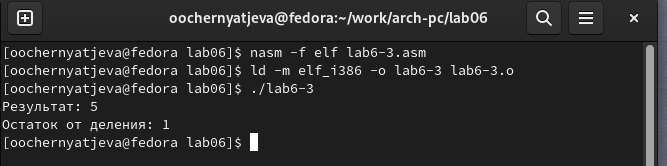


Figure 19: Запуск исполняемого файла

**20**

Создаю файл variant.asm с помощью утилиты touch (рис. [[20](#fig:020)]).

Figure 20: Создание файла

Figure 20: Создание файла

**21**

Ввожу в файл текст программы для вычисления варианта задания по номеру студенческого билета (рис. [[21](#fig:021)]).

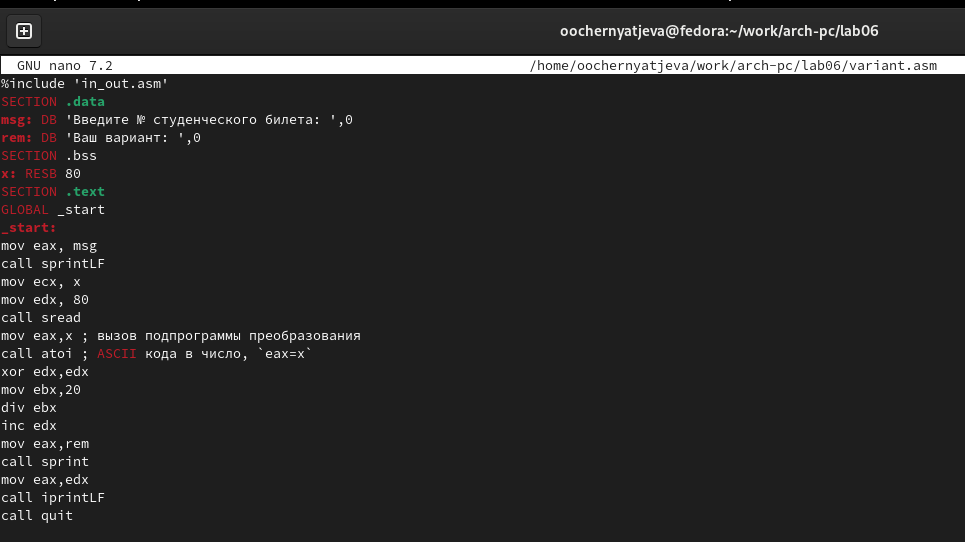


Figure 21: Редактирование файла

**22**

Создаю и запускаю исполняемый файл (рис. [[22](#fig:022)]). Ввожу номер своего студ. билета “1132239115” с клавиатуры, программа вывела, что мой вариант - 2.

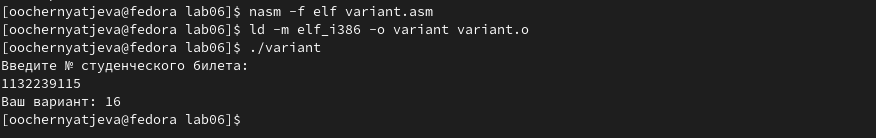


Figure 22: Запуск исполняемого файла

### 2.2.1 Ответы на вопросы по программе

1. За вывод сообщения “Ваш вариант” отвечают строки кода:

mov eax,rem  
call sprint

1. Инструкция mov ecx, x используется, чтобы положить адрес вводимой строки x в регистр ecx mov edx, 80 - запись в регистр edx длины вводимой строки call sread - вызов подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры
2. call atoi используется для вызова подпрограммы из внешнего файла, которая преобразует ascii-код символа в целое число и записывает результат в регистр eax
3. За вычисления варианта отвечают строки:

xor edx,edx ; обнуление edx для корректной работы div  
mov ebx,20 ; ebx = 20  
div ebx ; eax = eax/20, edx - остаток от деления  
inc edx ; edx = edx + 1

1. При выполнении инструкции div ebx остаток от деления записывается в регистр edx
2. Инструкция inc edx увеличивает значение регистра edx на 1
3. За вывод на экран результатов вычислений отвечают строки:

mov eax,edx  
call iprintLF

## 2.3 Выполнение заданий для самостоятельной работы

**1**

Создаю файл lab6-4.asm с помощью утилиты touch (рис. [[23](#fig:023)]).

Figure 23: Создание файла

Figure 23: Создание файла

**2**

Открываю созданный файл для редактирования, ввожу в него текст программы для вычисления значения выражения 2 варианта: f(x) = (10x-5)^2 (рис. [[24](#fig:024)]).

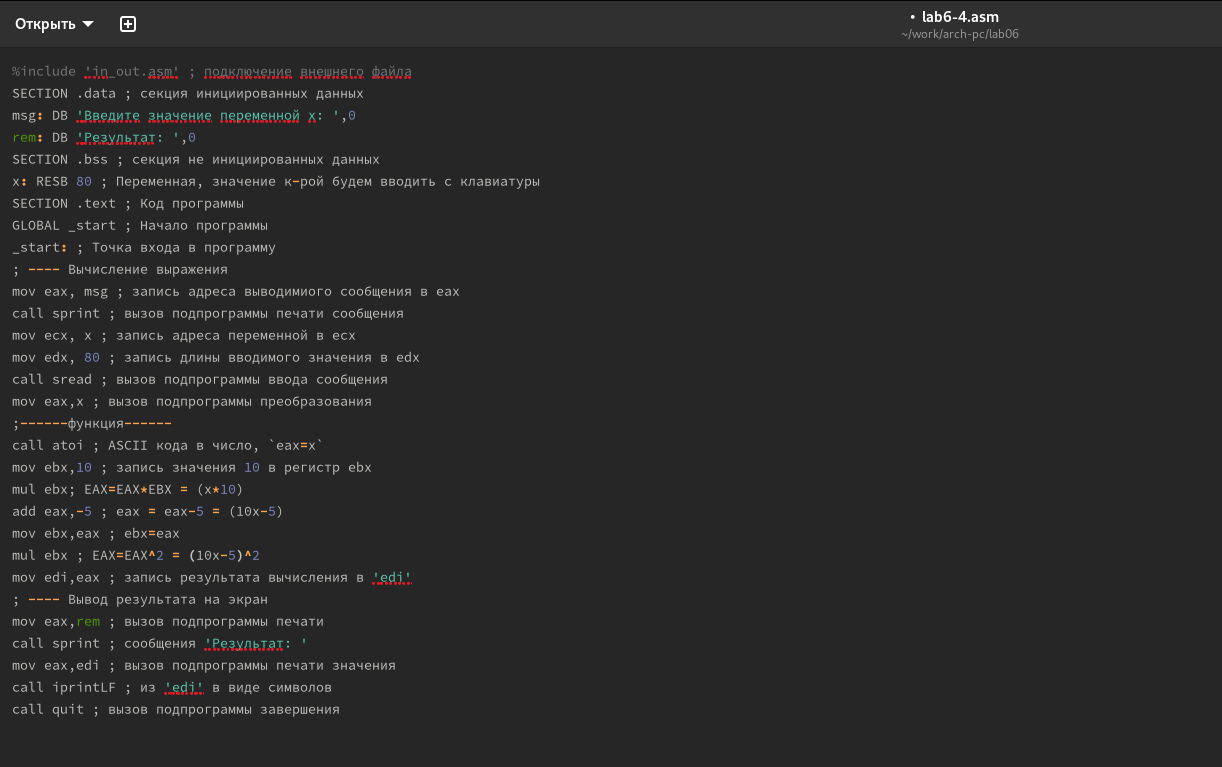


Figure 24: Написание программы

**3**

Создаю и запускаю исполняемый файл при вводе двух значений (рис. [[25](#fig:025)]). x = 3, f(x) = 625 Программа отработала верно.

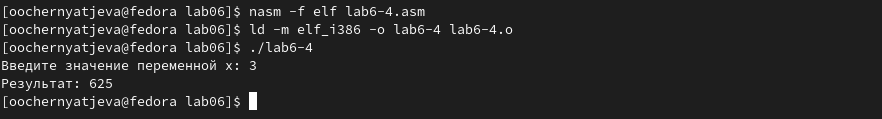


Figure 25: Запуск исполняемого файла

**4**

Запускаю исполняемый файл еще раз и ввожу второе значение икса x=1 для проверки на правильное вычисление результата. (рис. [[26](#fig:026)]). Программа отработала верно.



Figure 26: Запуск исполняемого файла

**Текст программы для вычисления значения выражения f(x)=(10x-5)^2**

%include 'in\_out.asm' ; подключение внешнего файла  
SECTION .data ; секция инициированных данных  
msg: DB 'Введите значение переменной х: ',0  
rem: DB 'Результат: ',0  
SECTION .bss ; секция не инициированных данных  
x: RESB 80 ; Переменная, значение к-рой будем вводить с клавиатуры  
SECTION .text ; Код программы  
GLOBAL \_start ; Начало программы  
\_start: ; Точка входа в программу  
; ---- Вычисление выражения  
mov eax, msg ; запись адреса выводимиого сообщения в eax  
call sprint ; вызов подпрограммы печати сообщения  
mov ecx, x ; запись адреса переменной в ecx  
mov edx, 80 ; запись длины вводимого значения в edx  
call sread ; вызов подпрограммы ввода сообщения  
mov eax,x ; вызов подпрограммы преобразования  
;------функция------  
call atoi ; ASCII кода в число, `eax=x`  
mov ebx,12 ; запись значения 2 в регистр ebx  
mul ebx; EAX=EAX\*EBX = (x\*12)  
add eax,3 ; eax = eax+3 = (3+12x)  
mov ebx,5 ; ebx =5  
mul ebx ; EAX=EAX\*EBX = (3+12x)\*5  
mov edi,eax ; запись результата вычисления в 'edi'  
; ---- Вывод результата на экран  
mov eax,rem ; вызов подпрограммы печати  
call sprint ; сообщения 'Результат: '  
mov eax,edi ; вызов подпрограммы печати значения  
call iprintLF ; из 'edi' в виде символов  
call quit ; вызов подпрограммы завершения

# 3 Выводы

При выполнении данной лабораторной работы я освоил арифметические инструкции: сложение, вычитание, умножение и деление на языке ассемблер NASM.