**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6**

*дисциплина: Архитектура компьютера*

Студент: Армихос Гонзалез Карла

Группа: НКАбд-02-24

**МОСКВА**

2024 г.

Содержание

[1. Цель работы 4](#_rk73iti44iis)

[2. Задание 5](#_excfirwaqflf)

[3. Технические введение 6](#_16umgbxblflt)

[4. Порядок выполнения лабораторной работы 8](#_427nfev9i8fe)

[4.1 Символьные и численные данные в NASM 8](#_k6isifw21m2p)

[4.2 Выполнение арифметических операций в NASM 11](#_lj07q2xle590)

[5. Задание для самостоятельной работы 16](#_s48tflb8idx8)

[6. Выводы 18](#_pv6euzc6a234)

[7. Список используемой литературы 18](#_g3c17rgwaije)

Список иллюстраций

### 

[рис. 4.1.1 Создание директории 8](#_3tjj6v1auxhz)

[рис. 4.2 Копи 4.1.2 8](#_95osf6go4kim)

[рис. 4.1.3 Запустить программу (j) 9](#_2hrru8izbpxl)

[рис. 4.1.4 Выполнить исправленную программу 9](#_ghvmhwpffbn8)

[рис. 4.1.5 Исправленный лист 9](#_bl356jgzdlgu)

[рис. 4.1.6 Новый файл lab6-2. asm 9](#_x83t72u6ya2y)

[рис. 4.1.7 Программа вывода значения регистра eax 10](#_vgvt2hdf5mqp)

[рис 4.1.8 Создайте исполняемый файл и запустите его. 10](#_387ar9ycscr9)

[рис 4.1.9 Измененная программа 11](#_fmi8joyypy8h)

[рис 4.1.10 Запустить программу 11](#_tinpanfwt9lg)

[рис. 4.1.11 Выполнить измененную функцию inprint 11](#_mcmj3gpsqjad)

[рис. 4.2.2 Новый файл lab6-3.asm 11](#_rxjgkagp17lj)

[рис. 4.2.3 Программа для вычисления арифметического выражения 12](#_vnt8cprvmzs)

[рис 4.2.4 Результат первой проблемы 12](#_l0taujq5l8j8)

[рис 4.2.5 Программа для вычисления арифметического выражения 13](#_3s8e5i5n3hhe)

[рис 4.2.6 Результат второй проблемы 13](#_mthqpyv1gkf1)

[Рис 4.2.7 Программу расчета варианта 14](#_1z07n7vwmjpo)

[Рис 4.2.8 Вариант 15](#_vdnvs4nvxvgm)

[рис 5.1 Программа 16](#_qczd7ex9hyk3)

[рис 5.2 Запустить программу 17](#_vkxouj657d01)

[рис 5.3 Запустить программу 17](#_a65tluisb16a)

### 

## 

# **1. Цель работы**

Освоение арифметических инструкций языка ассемблера NASM.

# 

# **2. Задание**

1. Работа с числовыми символами в NASM.
2. Запускать разные программы
3. Выполнение арифметических задач в NASM.
4. Выполнять работу самостоятельно

# **3. Технические введение**

В ассемблере команда сложения называется **ADD.**

В ассемблере команда сложения называется **ADD**. Синтаксис выглядит так:

**ADD** destination, source

Здесь destination — это регистр или память, куда будет записан результат, а source — это регистр, память или константа, которые прибавляются к destination.

Команда для умножения без знака называется **MUL**. Она используется для умножения операндов, рассматриваемых как положительные числа.

Команда для деления без знака — это **DIV.** Синтаксис деления чисел:

**DIV** divisor

Здесь **divisor** — это делитель. Делимое обычно находится в регистрах **AX**, **DX:AX**, или в регистрах шириной больше 8 бит.

При умножении двухбайтовых операндов (например, **MUL** с 16-битными значениями) результат записывается в регистры **DX:AX**, где **DX** содержит старшие 16 бит, а **AX** — младшие 16 бит результата.

Арифметические команды с целочисленными операндами и их назначение:

**ADD** — сложение целых чисел.

**SUB** — вычитание целых чисел.

**MUL** — умножение без знака.

**IMUL** — умножение со знаком.

**DIV** — деление без знака.

**IDIV** — деление со знаком.

**INC** — увеличение на 1 (инкремент).

**DEC** — уменьшение на 1 (декремент).

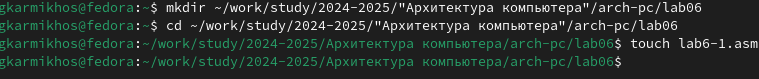
Местоположение делимого при целочисленном делении операндов:

Для целочисленного деления без знака (**DIV**), делимое должно находиться в регистрах **AX** (для 8-битного деления) или **DX:AX** (для 16-битного деления), где **DX** содержит старшие биты, а **AX** — младшие биты делимого.

# **4. Порядок выполнения лабораторной работы**

## 4.1 Символьные и численные данные в NASM

С помощью команды mkdir создайте каталог работы № 6, перейдите в него с команды cd и создайте файл lab6-1.asm: (рис. 4.1.1)



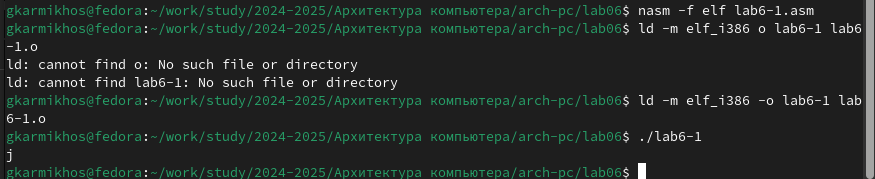
### рис. 4.1.1 Создание директории

В файле 6-1. asm напишите программу вывода значения регистра eax (рис. 4.1.2)



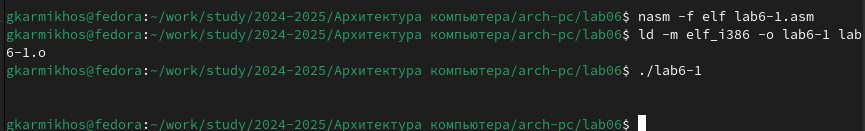
### рис. 4.2 Копи 4.1.2

Создайте исполняемый файл и запустите его. В этом случае при отображении значения регистра eax результатом будет символ j. (рис.4.1.3)



### рис. 4.1.3 Запустить программу (j)

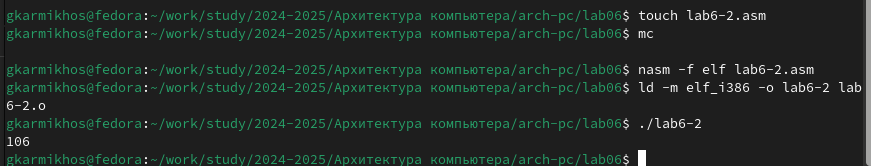
Изменить текст программы: mov eax,’6’ → mov eax,6 и mov ebx’4’ → mov ebx,4 (рис 4.1.5). В результате изменения кода при запуске программы возникает пробел (рис. 4.1.4).



### рис. 4.1.4 Выполнить исправленную программу

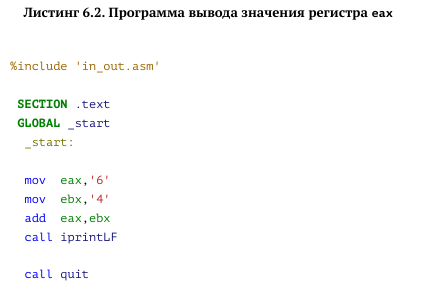
### рис. 4.1.5 Исправленный лист

Мы создаем новый файл с именем lab6-2. asm с помощью команды touch (рис. 4.1.6)



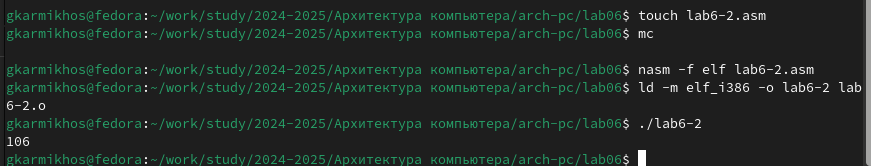
### рис. 4.1.6 Новый файл lab6-2. asm

Скопируйте следующую программу вывода значения регистра eax в наш lab6-2. asm (рис. 4.1.7)



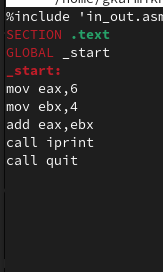
### рис. 4.1.7 Программа вывода значения регистра eax

Результат выполнения программы дал нам 106, поскольку команда add суммирует коды символов (рис 4.1.8)

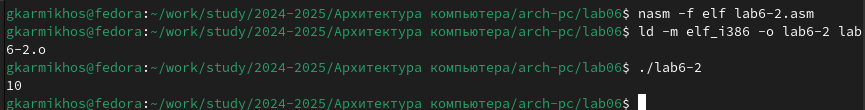


### рис 4.1.8 Создайте исполняемый файл и запустите его.

Мы редактируем код, удаляя кавычки (рис 4.1.9), и результат нашего кода дает нам результат 10 (рис 4.1.10).

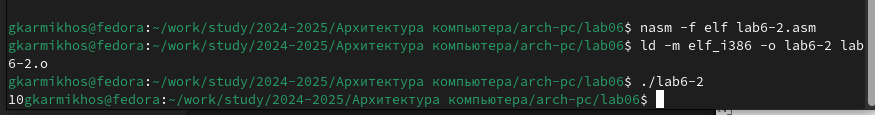


### рис 4.1.9 Измененная программа

****

### рис 4.1.10 Запустить программу

Меняем функцию inprintLF на inprint (рис. 4.1.11)

****

### рис. 4.1.11 Выполнить измененную функцию inprint

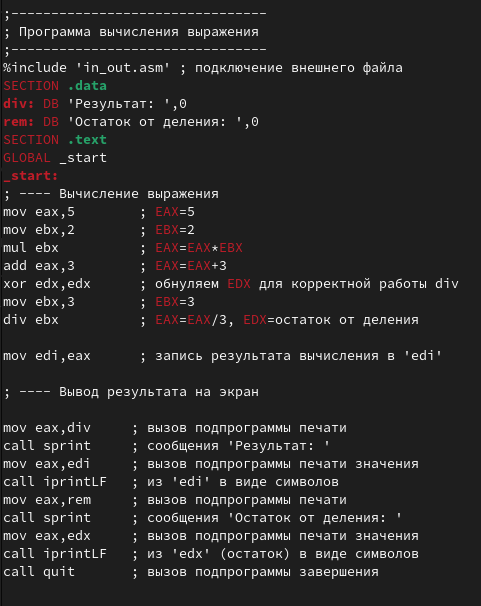
## 4.2 Выполнение арифметических операций в NASM

Создайте новый файл lab6-3.asm с помощью touch(рис. 4.2.1)



### рис. 4.2.2 Новый файл lab6-3.asm

Программа для вычисления арифметического выражения 𝑓(𝑥) = (5∗2+3)/3. (рис. 4.2.3)



### рис. 4.2.3 Программа для вычисления арифметического выражения

Результат арифметической задачи равен 4(рис 4.2.4)

### 

### рис 4.2.4 Результат первой проблемы

Решите следующую арифметическую задачу 𝑓(𝑥) = (4 ∗ 6 + 2)/5 (рис 4.2.5)

### 

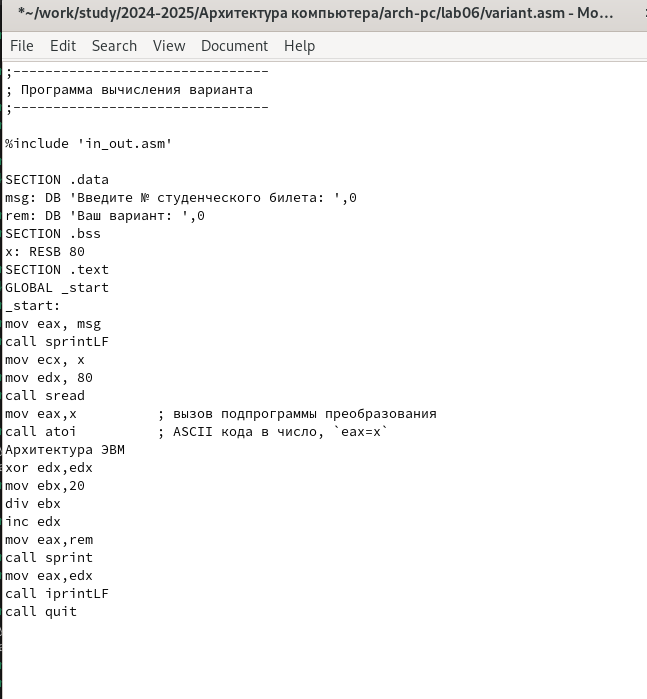
### рис 4.2.5 Программа для вычисления арифметического выражения

При выполнении программы мы получаем результат 5 (рис 5.2.6)

### 

### рис 4.2.6 Результат второй проблемы

Создайте файл variant.asm и написать программу расчета варианта задания по номеру студенческого билета (рис 4.2.7)



### Рис 4.2.7 Программу расчета варианта

Написать студенческий билет. Когда я писал свой студенческий билет, вариант был 16 (рис 4.2.8)

### Рис 4.2.8 Вариант

# 

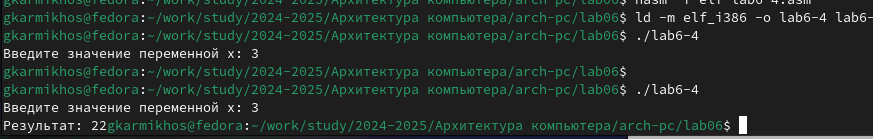
# **5. Задание для самостоятельной работы**

Написать программу вычисления выражения 𝑦 = 𝑓(𝑥) (рис 5.1).

### 

### рис 5.1 Программа

Вводим число и запускаем программу (рис 5.2)



### рис 5.2 Запустить программу

Вводим число и запускаем программу (рис 5.3)



### рис 5.3 Запустить программу

### 

# **6. Выводы**

В общем, **mc** - это инструмент управления файлами в терминальном режиме, позволяющий выполнять ряд обычных операций, таких как копирование, перемещение и удаление файлов, как с помощью команд bash, так и с помощью определенных комбинаций клавиш. Что касается языка ассемблера NASM, то его структура состоит из ключевых секций (**.data, .bss** и **.text**), которые позволяют определять различные типы данных и кода, оптимизируя использование памяти.

В этой лаборатории мы смогли познакомиться с работой NASM и Midnight Commander.

Зная функции клавиш F (1, 2, 5...10), мы смогли получить доступ к программе и работать над ней, а также использовать команды, ранее использовавшиеся в других лабораториях, такие как touch, nano.

# **7. Список используемой литературы**

1. [Архитектура ЭВМ](https://esystem.rudn.ru/pluginfile.php/2089086/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%966.%20%D0%90%D1%80%D0%B8%D1%84%D0%BC%D0%B5%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5%20%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%B8%20%D0%B2%20NASM..pdf)