**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЁТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №6**

**Дисциплина: Архитектура компьютера**

Студент: Ванюшкина Т.В.

Группа: НКАбд-01-24

Студ.билет: 1132246713

**МОСКВА**

**2024г**

**СОДЕРЖАНИЕ**

**1. Цель работы…………………………………………………….……………….3**

**2. Теоретическое введение………………………………….………………4**

**3. Выполнение лабораторной работы……………….……………5-14**

**3.1 Символьные и численные данные в NASM……………….5-9**

**3.2 Выполнение арифметических операций в NASM…….9-12**

**3.3 Задание для самостоятельной работы……………………13-14**

**4. Выводы………………………………………………………………..………….15**

**5. Список литературы………………………………………………………….16**

1. **Цель работы**

Целью работы является освоение арифметических инструкций языка ассемблера NASM.

1. **Теоретическое введение**

Язык ассемблера NASM (Netwide Assembler) - это мощный инструмент для работы с аппаратным обеспечением и создания низкоуровневого кода. В основе его возможностей лежат арифметические инструкции, которые позволяют выполнять основные математические операции над данными. В этом докладе мы рассмотрим ключевые арифметические инструкции NASM, их синтаксис, примеры использования и нюансы.

1. **Выполнение лабораторной работы**
   1. **Символьные и численные данные в NASM**

Создаю каталог для программам лабораторной работы № 6, перехожу в него и cоздаю файл lab6-1.asm:

Создание каталога

Рис.1: Создание каталога

Создание файла

Рис.2: Создание файла

Ввожу в файл lab6-1.asm текст программы из листинга 6.1.:

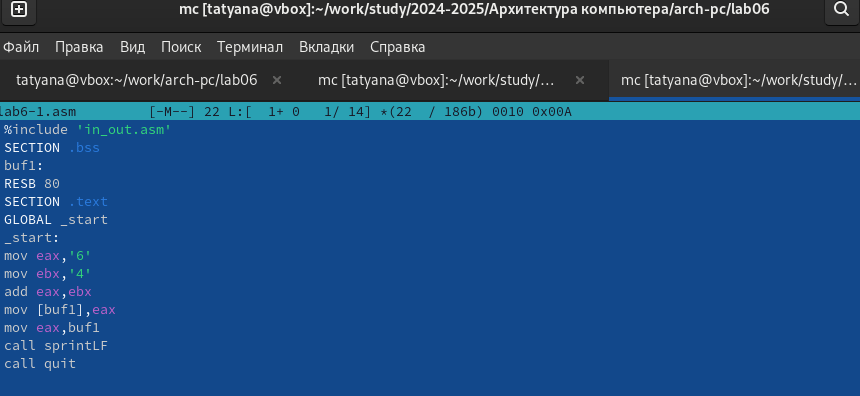


Рис 3:Ввод текста программы

Создаю исполняемый файл и запускаю его:

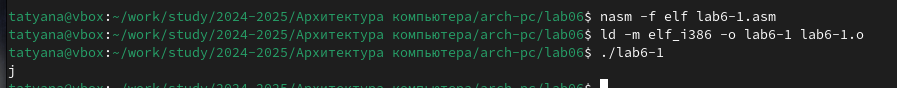


Рис.4:Создание и запуск файла

Исправляю текст программы (Листинг6.1) следующим образом: заменяю строки:

mov eax,'6'

mov ebx,'4'

на строки :

mov eax,6

mov ebx,4

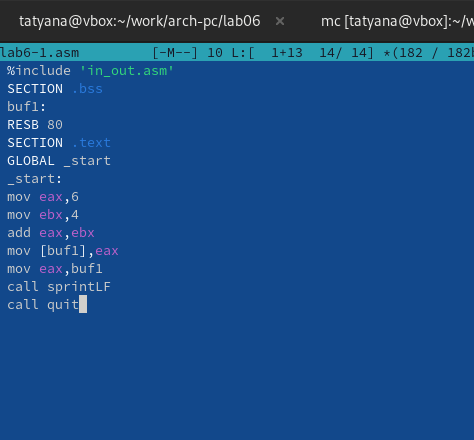


Рис.5: Замена текста программы

Создаю исполняемый файл и запускаю его:

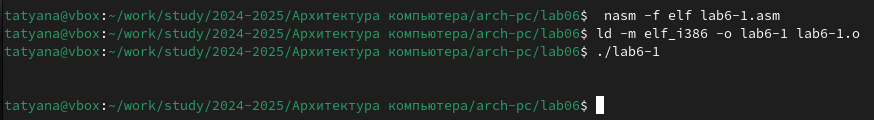


Рис.6 : Создание и запуск файла

Создаю файл lab6-2.asm:

Создание файла

Рис.7:Создание файла

Ввожу в него текст программы из листинга 6.2.:

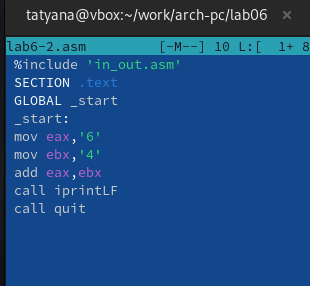


Рис.8: Ввод текста программы

Создаю исполняемый файл и запускаю его:

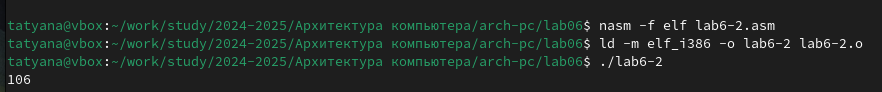


Рис.9: Создание и запуск файла

Аналогично предыдущему примеру изменяю символы на числа. Заменяю строки:

mov eax,'6'

mov ebx,'4'

на строки:

mov eax,6

mov ebx,4

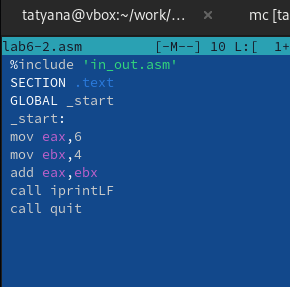


Рис.10: Замена текста программы

Создаю исполняемый файл и запускаю его:

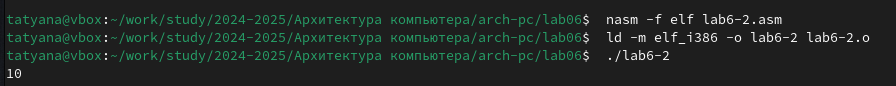


Рис.11: Создание и запуск файла

Заменяю функцию iprintLF на iprint:

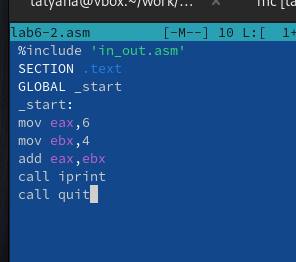


Рис.12: Замена текста программы

Создаю исполняемый файл и запускаю его:

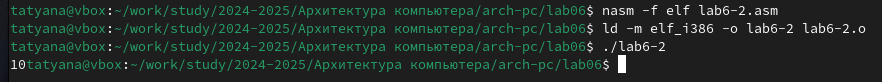


Рис.13: Создание и запуск файла

* 1. **Выполнение арифметических операций в NASM**

Создаю файл lab6-3.asm:

Создание файла

Рис.14: Создание файла

Ввожу в новый файл текст прогрммы из листинга 6.3:

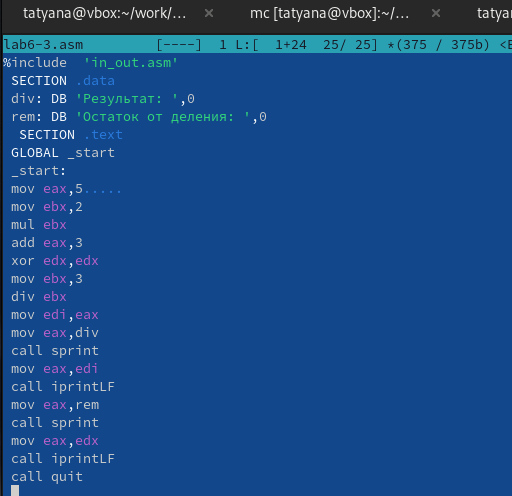


Рис.15: Ввод текста программы

Создаю исполняемый файл и запускаю его:

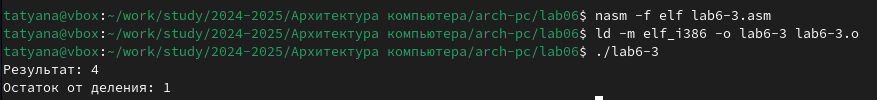


Рис.16: Создание и запуск файла

Изменяю текст программы для вычисления 𝑓(𝑥) = (4 ∗ 6 + 2)/5:

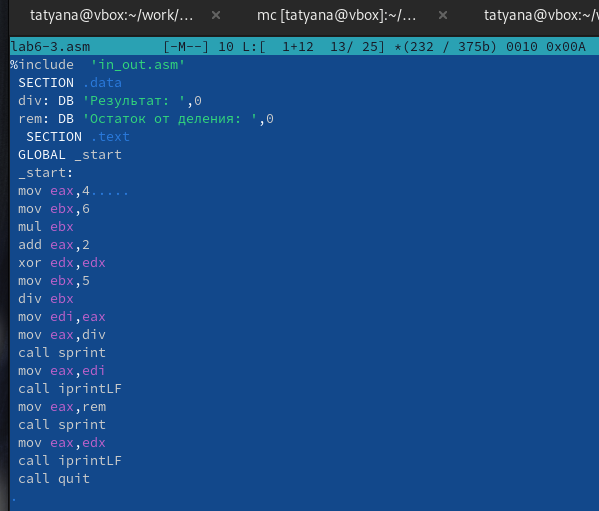


Рис.17: Замена текста программы

Создаю исполняемый файл и проверяю его работу:

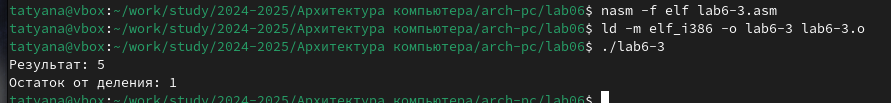


Рис.18: Создание и проверка файла

Создаю файл variant.asm:

Создание файла

Рис.19: Создание файла

Ввожу в новый файл текст из листинга 6.4 :

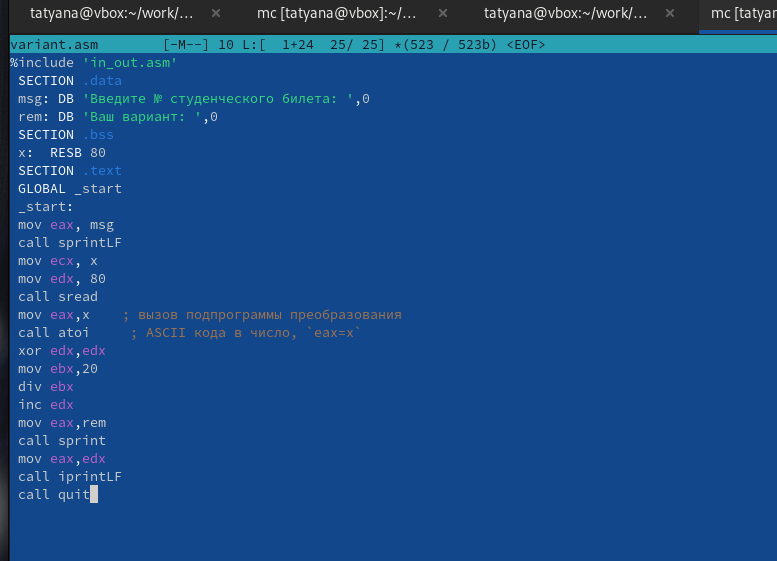


Рис.20: Ввод текста файла

Создаю исполняемый файл и запускаю его:

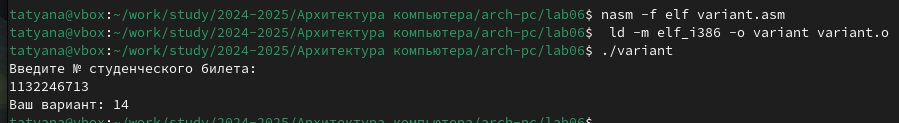


Рис.21: Создание и запуск файла

Ответы на вопросы:

1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’?

ответ: за вывод на экран сообщения "Ваш вариант" отвечают строки

mov eax,rem

call sprint

1. Для чего используется следующие инструкции?

mov ecx, x

mov edx, 80

call sread

ответ: инструкция mov ecx, x используется для того, чтобы положить адрес вводимой строки x в регистр

инструкция ecx mov edx, 80 - это инструкция для записи в регистр edx длины вводимой строки

инструкция call sread - вызывает подпрограмму из внешнего файла, обеспечивающую ввод сообщения с клавиатуры

1. Для его используется инструкция “call atoi”?

ответ: инструкция call atoi используется для вызова подпрограммы из внешнего файла, которая преобразует ascii-код символа в целое число и записывает результат в регистр eax

1. Какие строки листинга 6.4 отвечаютза вычисления варианта?

ответ: 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. В какой регистр записывается остаток отделения при выполнении инструкции“div ebx”?

ответ: остаток от деления записывается в регистр edx

1. Для чего используется инструкция“inc edx”?

ответ: инструкция inc edx увеличивает значение регистра edx на 1

1. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений?

ответ: за вывод на экран результата вычислений отвечают строки: mov eax,edx call iprintLF

* 1. **Задание для самостоятельной работы**

В каталоге создаю файл lab6-4.asm:

Создание файла

Рис.22: Создание файла

Создаю программу для вычисления варианта:

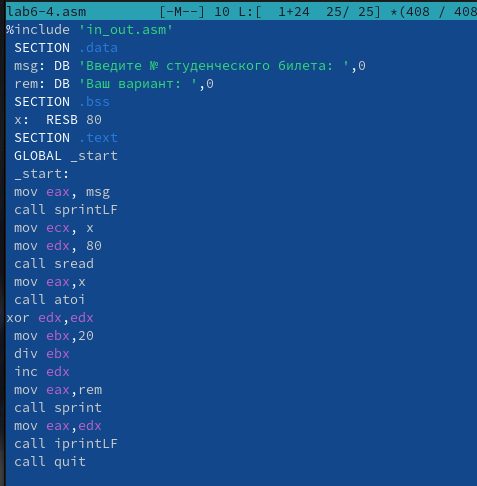


Рис.23: Создание программы

Создаю исполняемый файл и запускаю его:

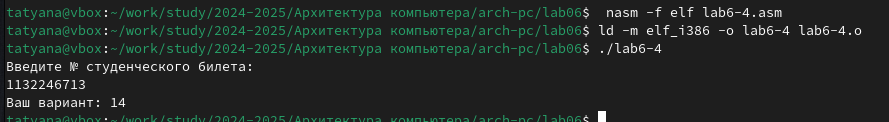


Рис.24: Создание и запуск файла

Создаю файл variant14.asm:

Создание файла

Рис.25: Создание файла

Ввожу в него текст программы для вычисления выражения 14) (𝑥/2+8)⋅3:

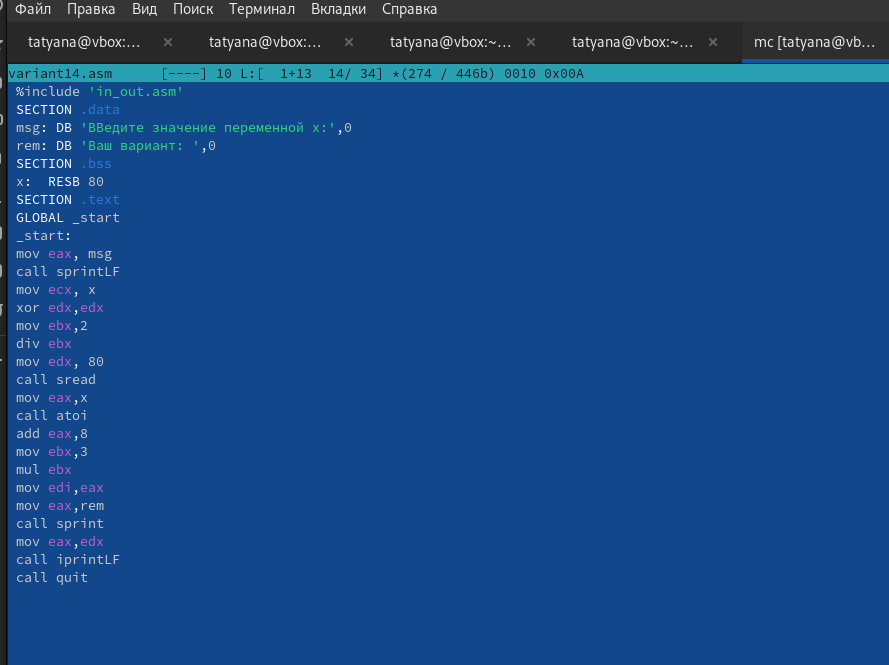


Рис.26: Ввод текста программы

Создаю исполняемый файл и запускаю его:

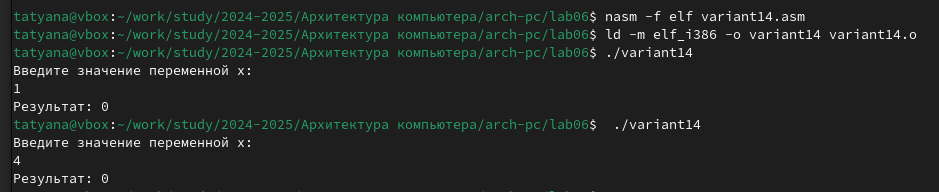


Рис.27: Создание и запуск файла

1. **Выводы**

В ходе выполнения лабораторной работы я освоила арифметические инструкции языка ассемблера NASM.

1. **Список литературы**

Курс: Архитектура компьютеров и операционные системы. Раздел "Архитектура компьютеров" (02.03.00, УГСН) (rudn.ru)