Отчёт по лабораторной работе №6

Дисциплина: Архитектура компьютера

Будник Александра Олеговна

Содержание

# 1 Цель работы

Приобрести навыки работы с арифметическими инструкциями языка ассемблера NASM.

# 2 Теоретическое введение

Большая часть инструкций на языке ассемблера требует обработки операндов. Адрес операнда - это место, где хранятся данные, которые подлежат обработке.

Существует три основных способа адресации: 1. Регистровая адресация – операнды хранятся в регистрах и в команде используются имена этих регистров, например: mov ax,bx. 2. Непосредственная адресация – значение операнда задается непосредственно в ко- манде, Например: mov ax,2. 3. Адресация памяти – операнд задает адрес в памяти. В команде указывается символи- ческое обозначение ячейки памяти.

Table 1: Описание некоторых команд на языке ассемблера

| Команда | Арифметическая операция |
| --- | --- |
| add | Целочисленное сложение |
| sub | Целочисленное вычитание |
| neg | Изменение знака операнда |
| mul | Умножение |
| div | Деление |

# 3 Выполнение лабораторной работы

При помощи команды mkdir создаю папку для работы (рис. ??).

Создание директории

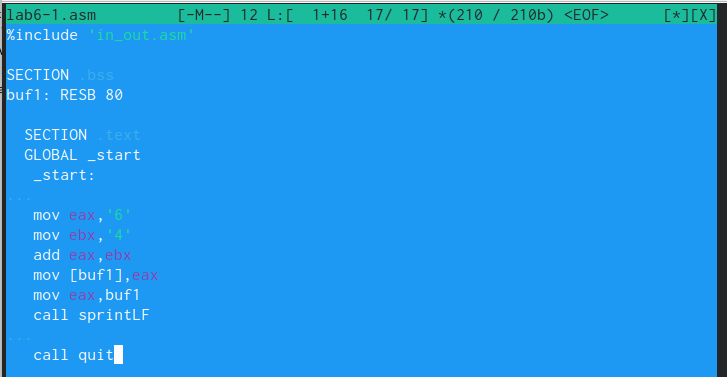
Создание директории

Перехожу в созданный каталог ~/work/arch-pc/lab06 и создаю файл lab6-1.asm (рис. ??).

Создание файла

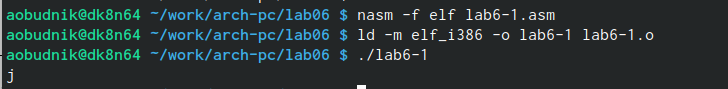
Создание файла

Ввожу код программы из листинга 6.1 для вывода значения регистра eax (рис. ??).



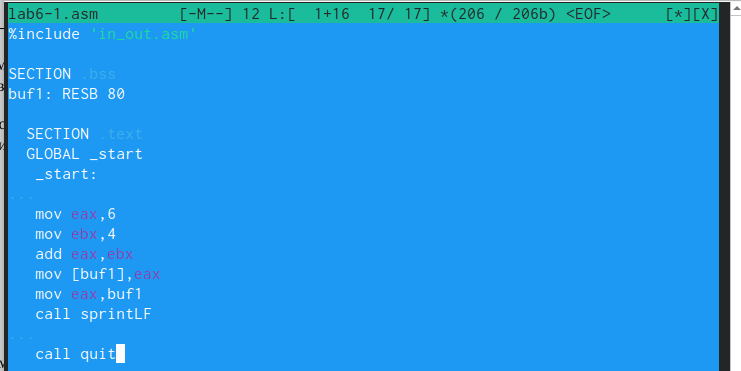
Код программы lab6-1

Создаю объектный файл и с помощью компоновщика LD обрабатываю его. Получаю исполняемый файл. Запускаю программу (рис. ??).



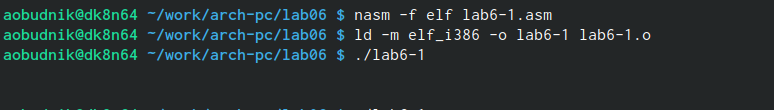
Запуск программы

Изменяю код в соответствии с заданием (рис. ??).



Изменение кода

Создаю объектный файл и с помощью компоновщика LD обрабатываю его. Получаю исполняемый файл. Запускаю программу. Полученный результат - символ перехода на новую строку (рис. ??).



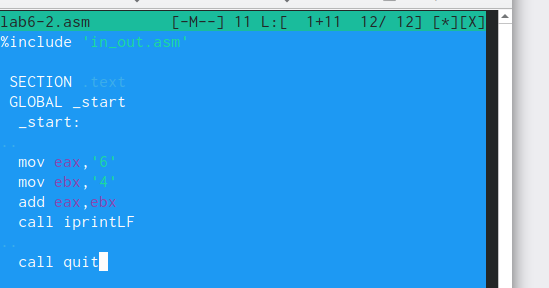
Запуск программы

Создаю файл lab6-2.asm (рис. ??).

Создание файла

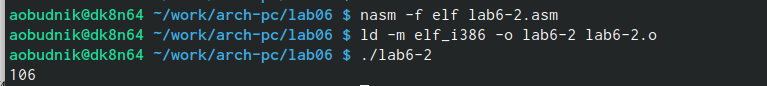
Создание файла

Ввожу код программы из листинга 6.2 для вывода значения регистра eax (рис. ??).



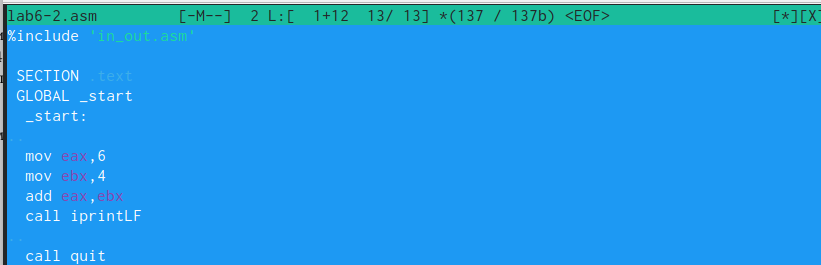
Код программы lab6-2

Создаю объектный файл и с помощью компоновщика LD обрабатываю его. Получаю исполняемый файл. Запускаю программу (рис. ??).



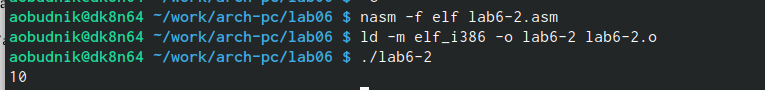
Запуск программы

Изменяю код программы, меняя символы на числа (рис. ??).



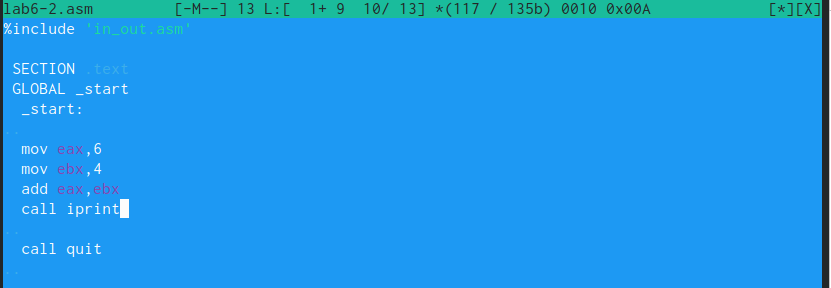
Изменение кода

Произвожу трансляцию и линковку. Запускаю получившуюся программу (рис. ??).



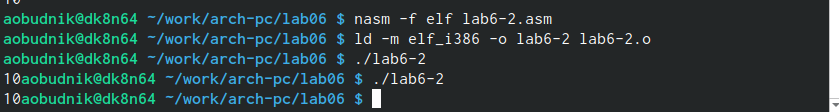
Запуск программы

Изменяю iprintLF на iprint в тексте программы. (рис. ??).



Изменение кода

Произвожу трансляцию и линковку. Запускаю получившуюся программу (рис. ??). Команда iprint не добавляет к выводу символ переноса строки, в отличие от iprintLF.



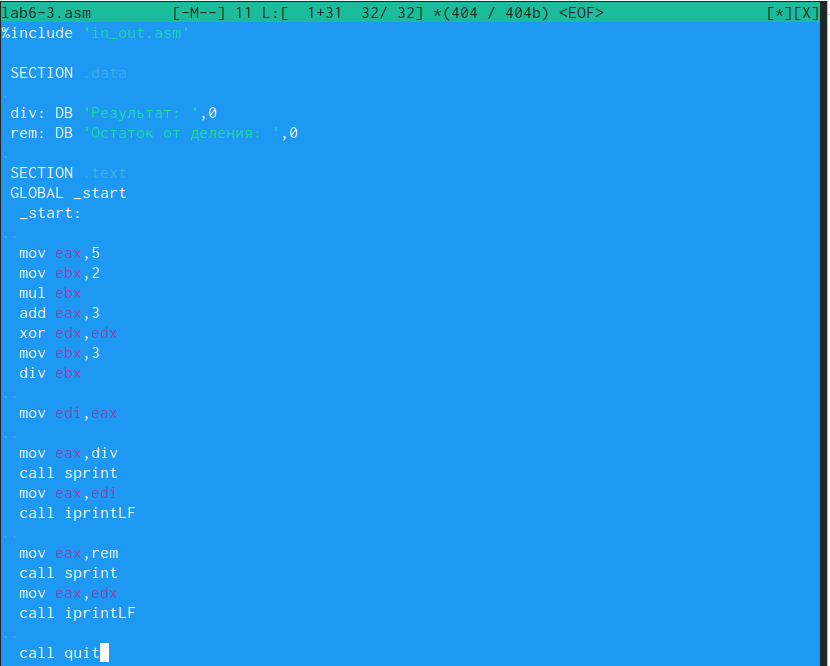
Запуск программы

В каталоге создаю файл lab6-3.asm (рис. ??).

Создание файла

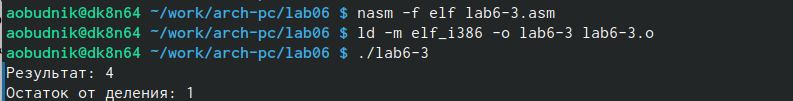
Создание файла

Ввожу код программы из листинга 6.3 для вычисления выражения 𝑓(𝑥) = (5 ∗ 2 + 3)/3 (рис. ??).



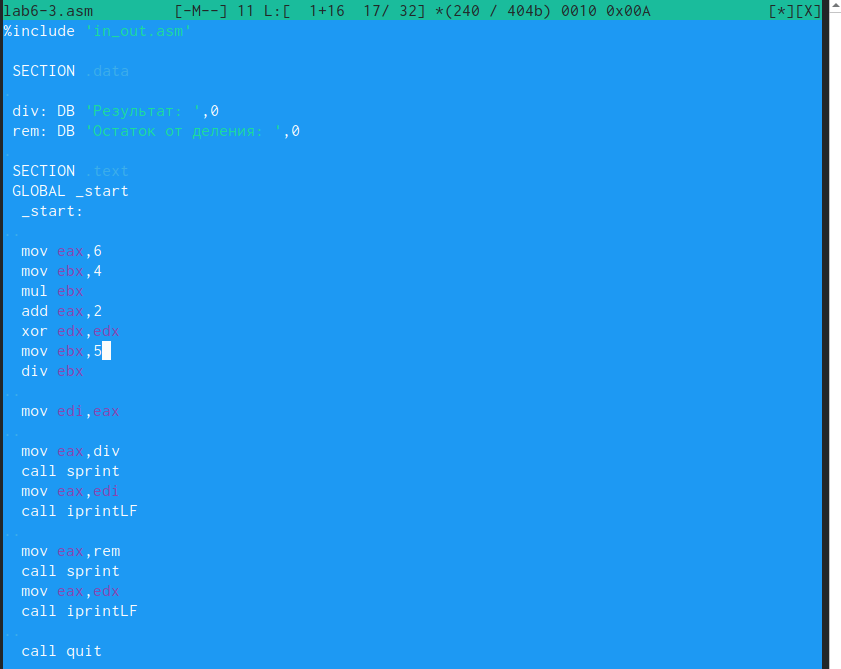
Код программы lab6-3

Создаю объектный файл и обрабатываю его при помощи компоновщика. Получаю исполняемый файл и запускаю программу (рис. ??).



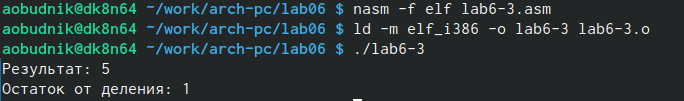
Запуск программы

Изменяю код программы так, чтобы она вычисляла значение выражения 𝑓(𝑥) = (4 ∗ 6 + 2)/5. (рис. ??).



Изменение кода

Произвожу трансляцию и линковку. Запускаю получившуюся программу (рис. ??).



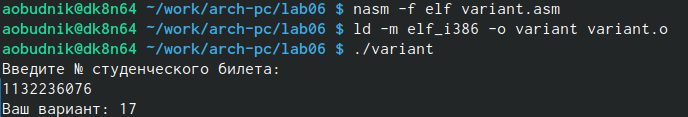
Запуск программы

Создаю файл variant.asm при помощи утилиты touch и ввожу в нее код программы из листинга 6.4 для вычисления вычисления варианта задания по номеру студенческого билета (рис. ??).



Код программы variant

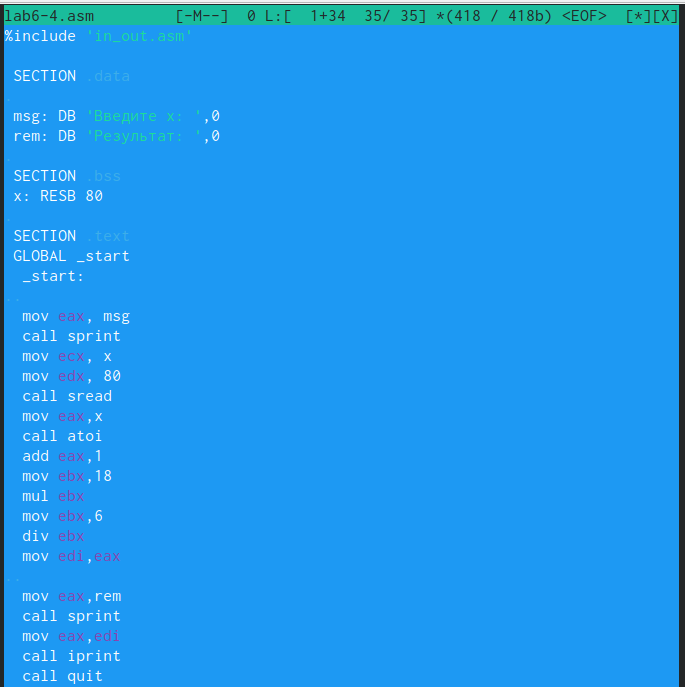
Произвожу трансляцию и линковку. Запускаю получившуюся программу (рис. ??). Мой вариант под номером 17.



Запуск программы

# 4 Выполнение заданий для самостоятельной работы

При помощи команды touch создаю файл lab6-4.asm. Ввожу код программы для вычисления выражения 18(𝑥 + 1)/6 относительно переменной х (рис. ??).



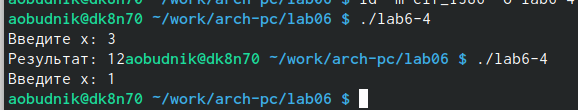
Код программы lab6-4

Создаю объектный файл и обрабатываю его при помощи компоновщика. Получаю исполняемый файл (рис. ??).

Линковка

Линковка

Запускаю программу и проверяю результат для значений x1 и x2 (рис. ??).



Запуск программы

Все созданные в процессе работы файлы копирую в локальный репозиторий и загружаю на GitHub.

# 5 Выводы

По итогам выполнения лабораторной работы №6 я научилась работать с некоторыми арифметическими инструкциями языка ассемблера NASM.