Лабораторная работа №6

НКАбд-02-23

Выборнов Дмитрий Валерьевич

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера **NASM**.

# 2 Задание

1. Символьные и численные данные в NASM.
2. Выполнение арифметических операций в NASM.
3. Вопросы по программе.
4. Задание для самостоятельной работы.

# 3 Теоретическое введение

Большинство инструкций на языке ассемблера требуют обработки операндов. Адрес операнда предоставляет место, где хранятся данные, подлежащие обработке. Это могут быть данные хранящиеся в регистре или в ячейке памяти. Три основных способа адресации: регистровая адресация, непосревственная адресация и адресация памяти. В **NASM** доступны команды сложения, вычетания, умножения, деления, изменения знака, инкремента и декремента. Ввод информации с клавиатуры и вывод её на экран осуществляется в символьном виде. Кодирование этой информации производится согласно кодовой таблице символов ASCII. ASCII – сокращение от American Standard Code for Information Interchange (Американский стандартный код для обмена информацией). Согласно стандарту ASCII каждый символ кодируется одним байтом.

# 4 Выполнение лабораторной работы

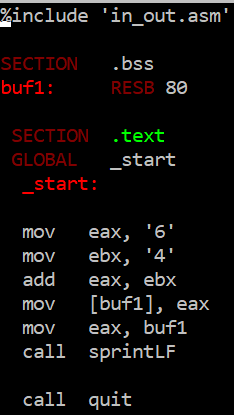
## 4.1 Символьные и численные данные в NASM.

Перехожу в каталог курса, создаю отдельный каталог для шестой лабораторной работы, перехожу в него и в нём создаю файл для первого задания.



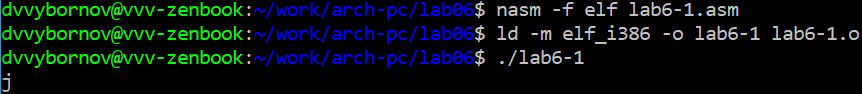
Первый шаг.

Ввожу текст программы вывода регистра eax.



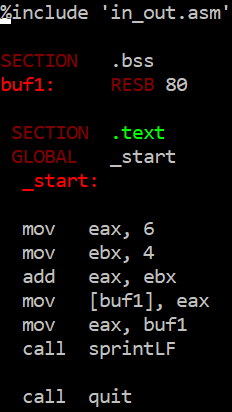
Второй шаг.

Создаю исполняемый файл и запускаю его.



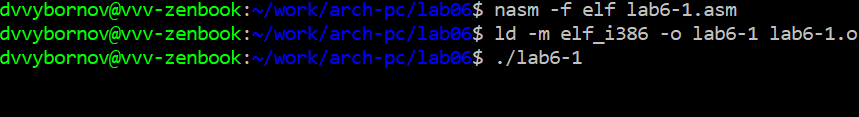
Третий шаг.

Изменяю текст программы.



Четвёртый шаг.

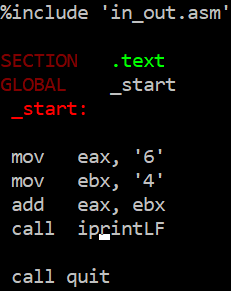
Запускаю изменённую программу.



Пятый шаг.

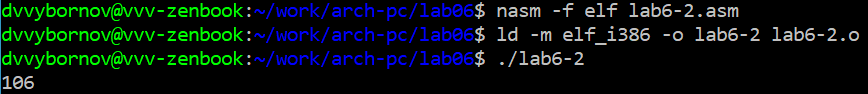
На экране ничего не отображается, так как 10 - это код символа перевода строки.

Создаю новый файл и ввожу в него другую версию программы вывода регистра eax.



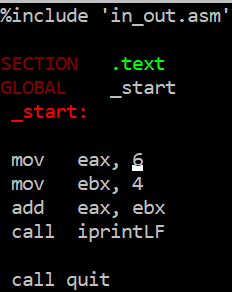
Шестой шаг.

Создаю исполняемый файл новой версии программы и запускаю его.



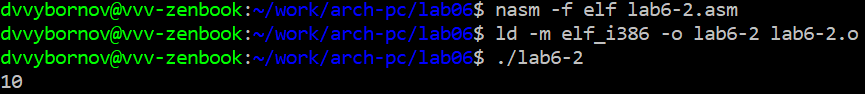
Седьмой шаг.

Изменяю новую версию программы аналогично предыдущей.



Восьмой шаг.

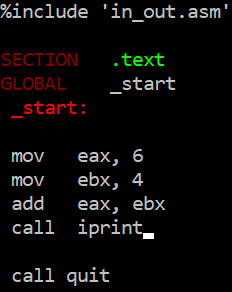
Запускаю изменённую программу.



Девятый шаг.

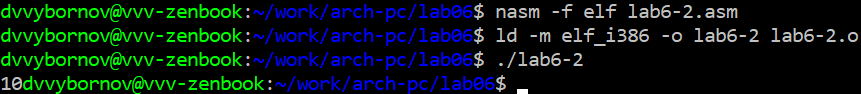
Программа выводит 10, т. к. теперь она складывает сами числа, а не их коды.

Заменяю функцию iprintLF на iprint.



Десятый шаг.

Запускаю полученную программу.

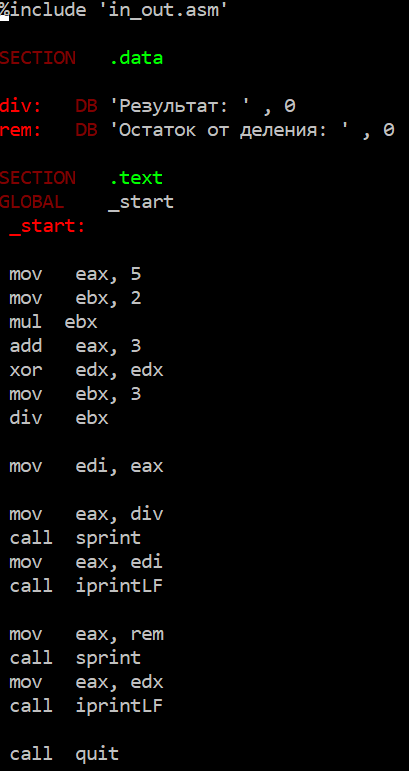


Одиннадцатый шаг.

Вывод функции iprintLF отличается от вывода функции iprint тем, что она добавляет к выводу символ переноса строки.

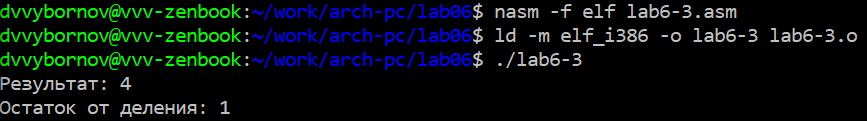
## 4.2 Выполнение арифметических операций в NASM.

Создаю новый файл lab6-3.asm и ввожу в него текст программы вычисления выражения.



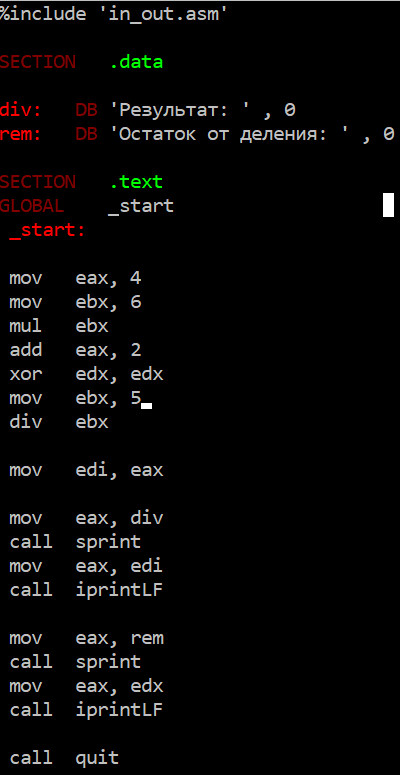
Двенадцатый шаг.

Создаю исполняемый файл и запускаю его.



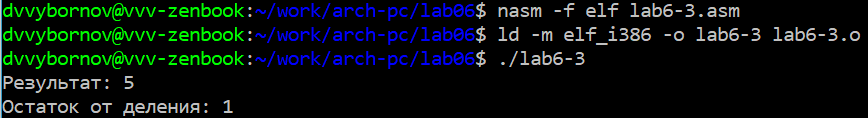
Тринадцатый шаг.

Изменяю текст программы для вычисления выражения F(x) = (4 \* 6 + 2)/5.



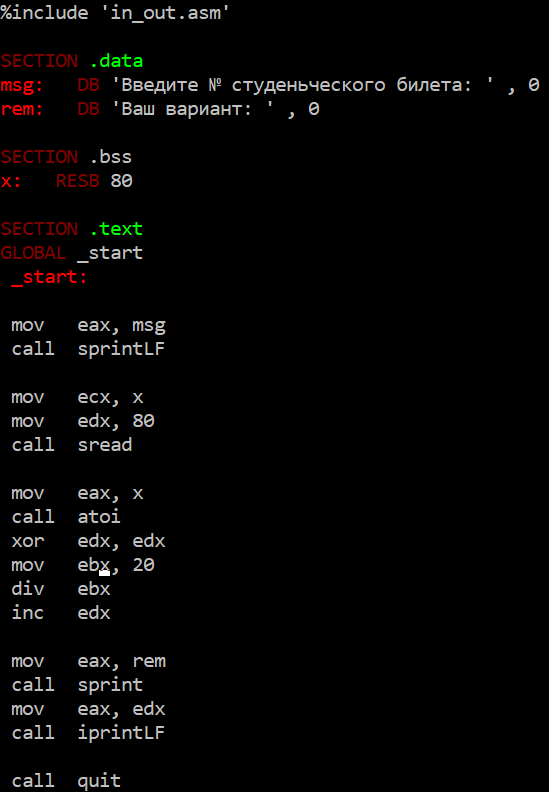
Четырнадцатый шаг.

Запускаю изменённую программу.



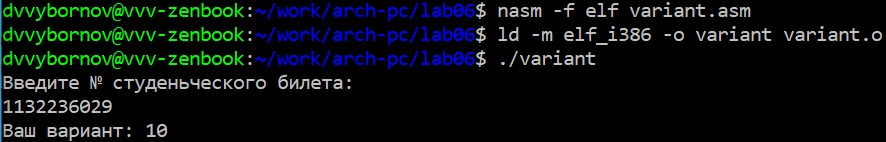
Пятнадцатый шаг.

Создаю новый файл и ввожу в него программу для вычисления варианта задания по номеру студенческого билета.



Шестнадцатый шаг.

Создаю исполняемый файл и запускаю его.



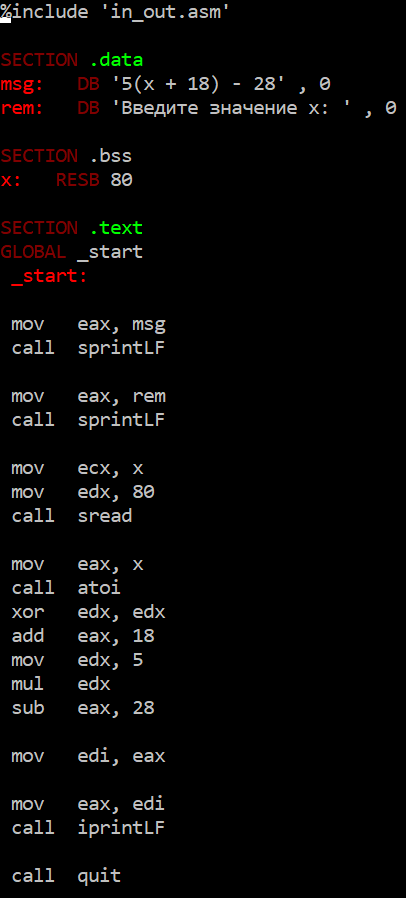
Семнадцатый шаг.

## 4.3 Ответы на вопросы по программе.

1. За вывод на экран сообщения “Ваш вариант:” отвечают строки **mov eax, rem** и **call sprint**.
2. Инструкция **mov eax, x** используется для определения адреса записи значения вводимой строки x в регистр eax. Инструкция **mov edx, 80** используется для записи длины вводимой строки в регистр edx, а инструкция **call sread** - для вызова подпрограммы из внешнего файла, которая используется для ввода с клавиатуры.
3. Строка **call atoi** используется для вызова подпрограммы из внешнего файла, которая преобразует ascii - код символа в целое число.
4. За вычисления варианта отвечают строки **xor edx, edx**, **mov edx, 20**, **div ebx** и **inc edx**.
5. Остаток записывается в регистр **edx**.
6. Инструкция **inc edx** увеличивает значение регистра **edx** на 1.
7. За вывод результата вычислений отвечают строки **mov eax, edc** и **call iprintLF**.

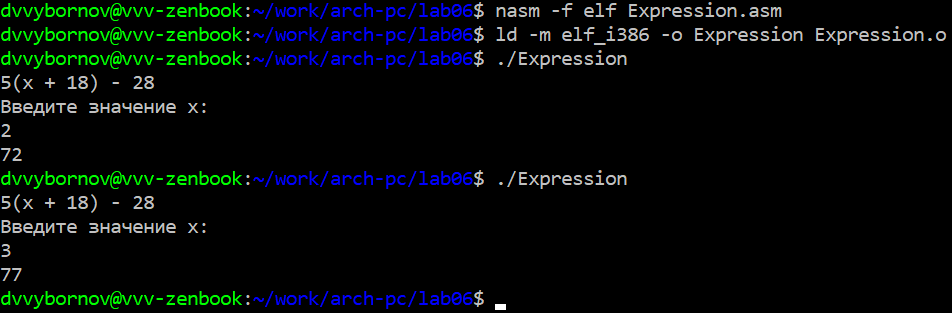
## 4.4 Задание для самостоятельной работы.

Создаю файл Expression.asm и ввожу в него текст программы для выражения №10.



Первый шаг задания для самостоятельной работы.

Создаю исполняемый файл, запускаю его и проверяю его работу для значений x = 2 и x = 3.



Второй шаг задания для самостоятельной работы.

# 5 Выводы

Выполнив эту лабораторную работу, я освоил арифметические инструкции языка ассемблера **NASM**.