Лабораторная работа №7.Арифметические

Дисциплина Архитектура ЭВМ

Осокин георгий иванович

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

# 2 Выполнение лабораторной работы

Создадим каталог и перейдем в него. Создадим lab7-1.asm

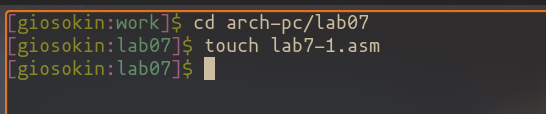


Рис. 1: Создание lab-1.asm

## 2.1 Складывание строк

Введем текст листинга в файл

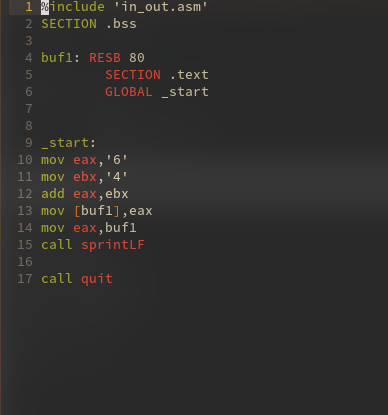


Рис. 2: Текст в lab-1.asm

Оттранслируем, слинкуем и запустим

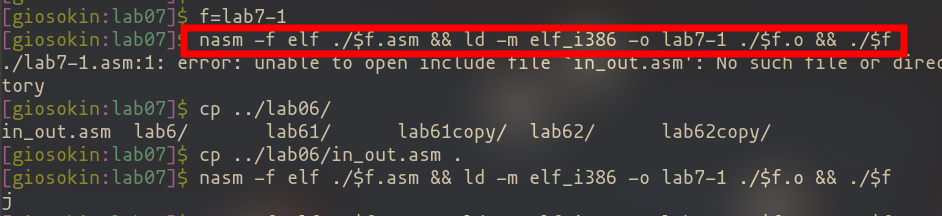


Рис. 3: Компиляция и запуск кода

Видим, что вывелось “j” потому что мы складываем не числа, а коды строк, и в результате выводим код строки “j”

## 2.2 Складывание чисел

Заменим строки на числа, скомпилируем и запустим

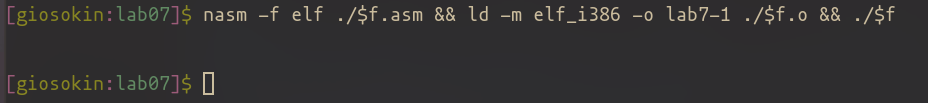


Рис. 4: Выполнение кода, с замененными строками на числа

Нам вывелся перенос строки, так как мы хоть и сложили числа, но так же вывели не число 10, а символ которому присвоен код 10, то есть перенос строки

## 2.3 Программа вывода значение регистра eax

Создадим фалй lab7-2.asm

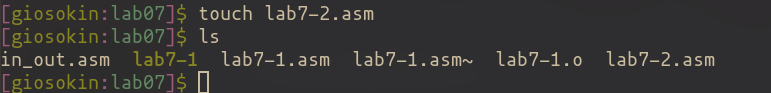


Рис. 5: Создание фалйа lab7-2.asm

Введем текст

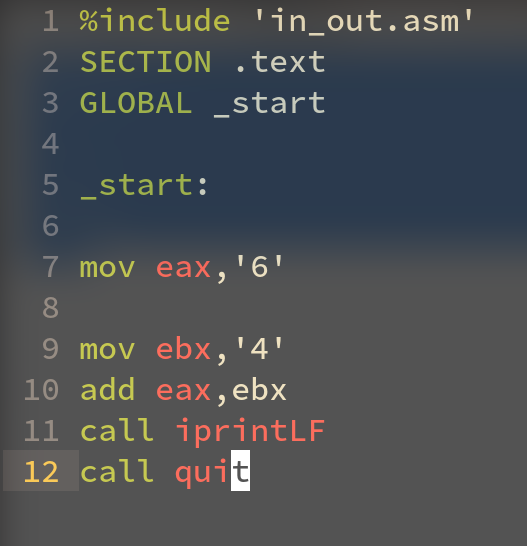


Рис. 6: Текст в lab7-2.asm

Запустим код

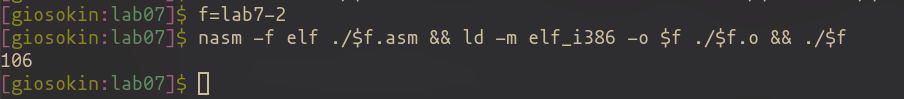


Рис. 7: Компиляция и запуск

Программа вывела 106, так как мы выводим аддрес символа, который мы получаем, суммируя адрессы двух строк

Заменим sprintLF на iprintLF и запустим

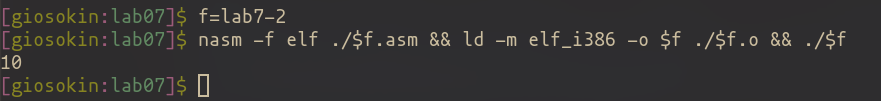


Рис. 8: Запуск измененной программы

Вывелось

Заменим iprintLF на iprint

Рис. 9: Выпоск кода с замененным iprintLF на iprint

Рис. 9: Выпоск кода с замененным iprintLF на iprint

Вывелось 10, без перевода строки

## 2.4 Программа для вычесления выражения

Создадим файл lab7-3.asm

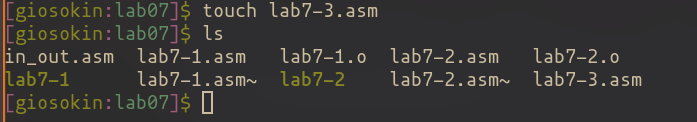


Рис. 10: Создание файла lab7-3.asm

Запишем текст в файл и запустим.

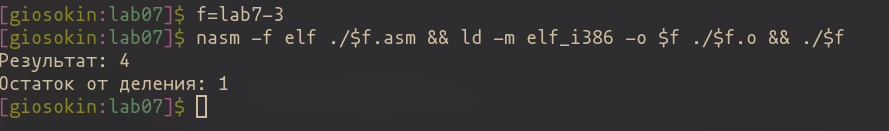


Рис. 11: Компиляция и запуск lab7-3.asm

## 2.5 Изменение программы для вычисления выражения

Заменим числа и исполним

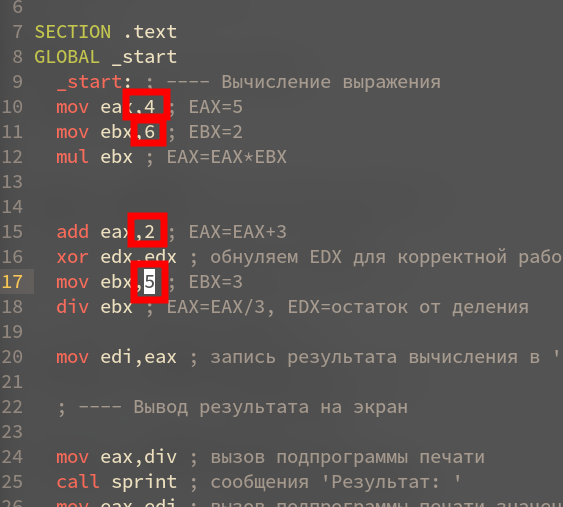


Рис. 12: Изменения кода для вычесления другой функции

Скомпилируем и запустим

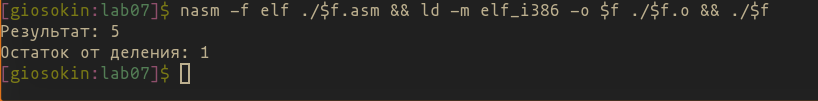


Рис. 13: Вывод измененный программы lab7-3.asm

## 2.6 Вычесление варианта по студенческому билеты

Создадим файл variant.asm с текстом листинга 7.4

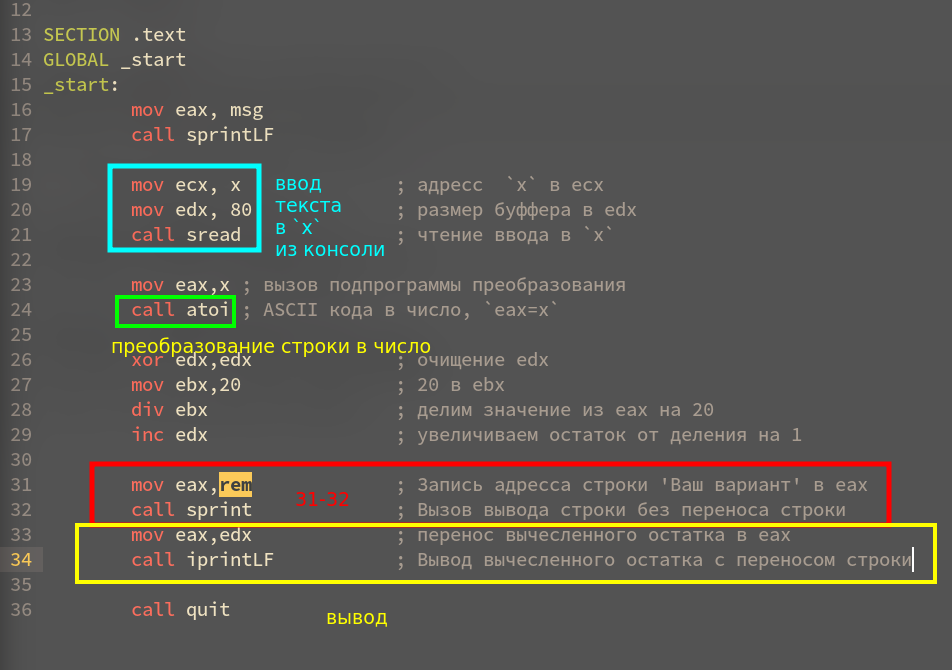


Рис. 14: Объяснение блоков кода

1. Для вывода на экран “Ваш вариант” отвечают стоки 31-32 (выделены красным)
2. инструкции на строка 19-21 (выделены голубым) используется для ввода текста в x из консоли
3. call atoi используется для преобразования ACII кода в число
4. За вычисление варианта отвечают строки 26-29
5. Остаток от деления записывается в edx/dx
6. inc edx увеличивает значение edx на 1
7. За вывод результата вычислений отвечают строки 33-34 (выделенно желтым)

Запустим программу что бы узнать номер варианта

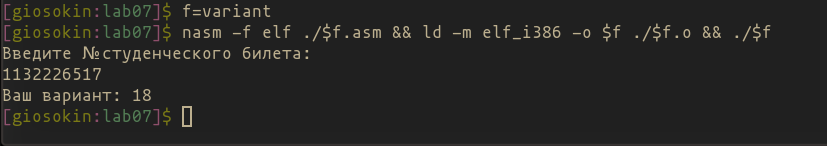


Рис. 15: Запуск программы по вычислению варианта

Наш варинт 18

# 3 Задания для самостоятельной работы

## 3.1 Написать программу по вычислению

Напишем, код, который будет вычислять.

Изменим размер буффера с 80 на 10, так как для чисел 80 байт многовато.

Так как введенное значение уже в eax просто добавим к нему 10, домножим на 3 и отнимем 20

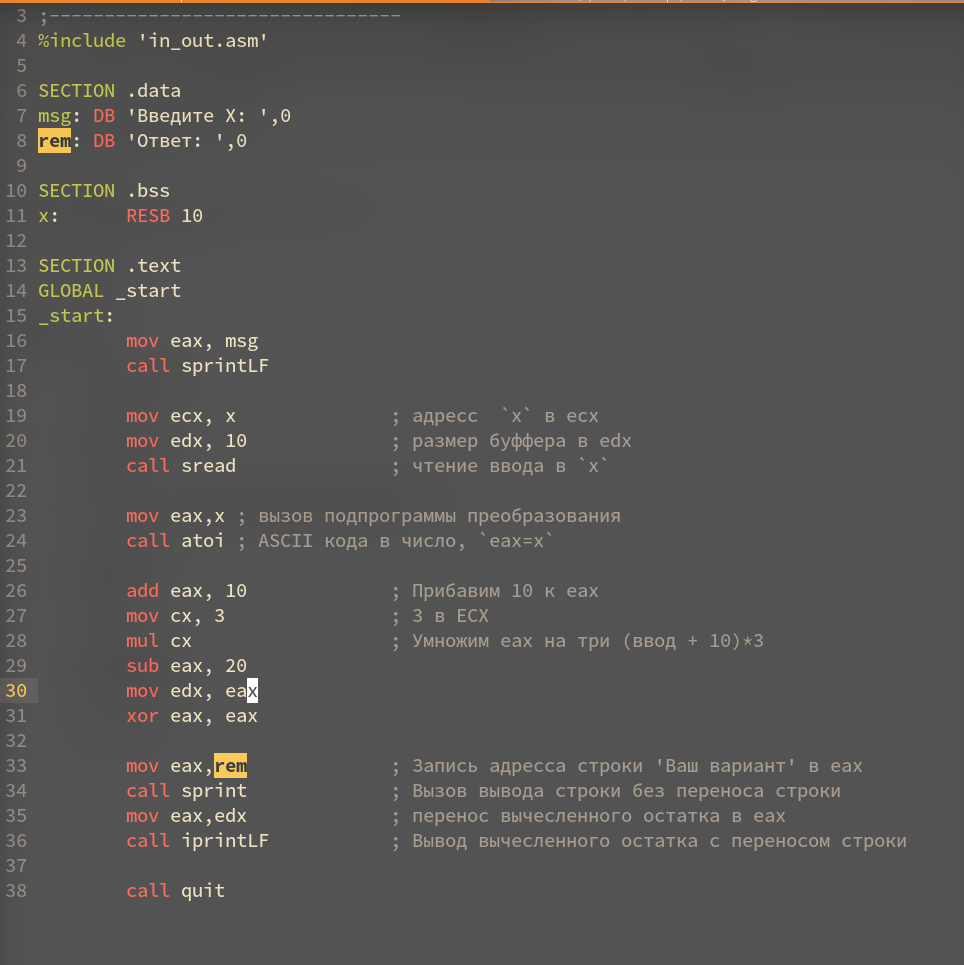


Рис. 16: Код, вычисления выражения 3(x+10) - 20

Запустим и убедимся в верности выполненных вычислений

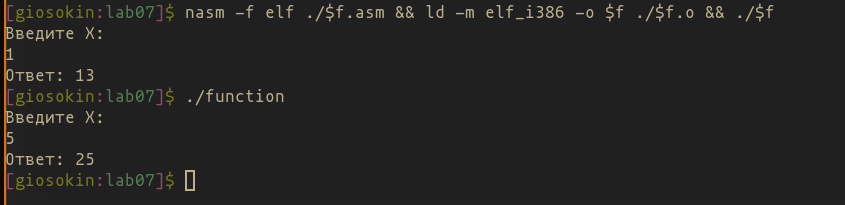


Рис. 17: Вывод вычислений

# 4 Выводы

Мы освоили арифмитические операции в NASM и написали программу, которое вычисляет значение функции от x