Лабораторная работа №8. Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Дисциплина: Архитектура компьютера

Алексей Назаров НММбд-02-22

Содержание

# 1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навы- ков написания программ с использованием переходов. Знакомство с назначением и стуруктурой файлов листинга

# 2 Выполнение лабораторной работы

## 2.1 Создадим каталог для лабораторной и файл lab8-1.asm

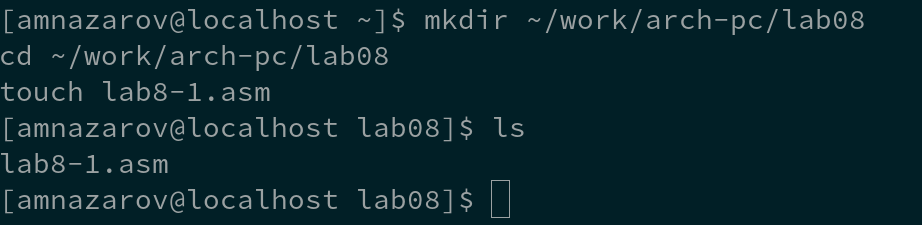


Рис. 1: Создадим файл и папку

## 2.2 Введем в файл lab8-1.asm текст из листинга

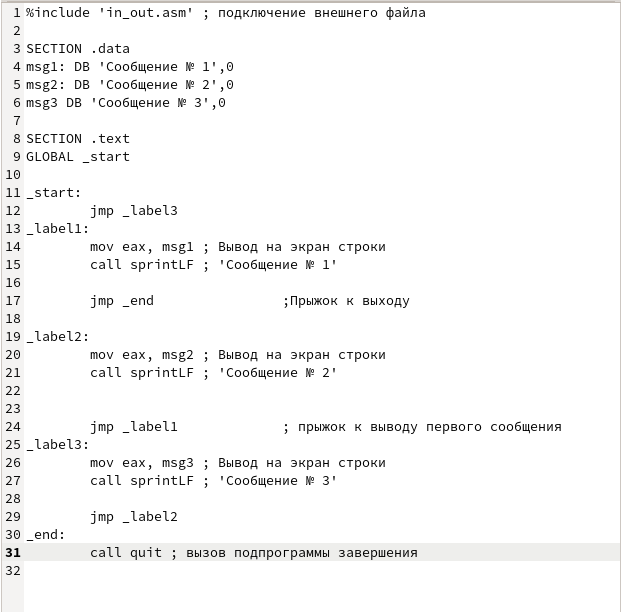


Рис. 2: Код lab8-1

Запустим программу и посмотрим результат

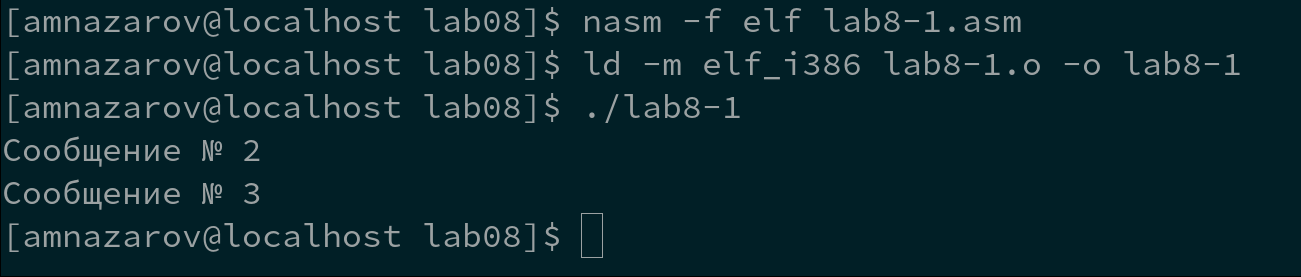


Рис. 3: Запуск lab8-1

## 2.3 Изменим код, как показанно в листинге 8.2

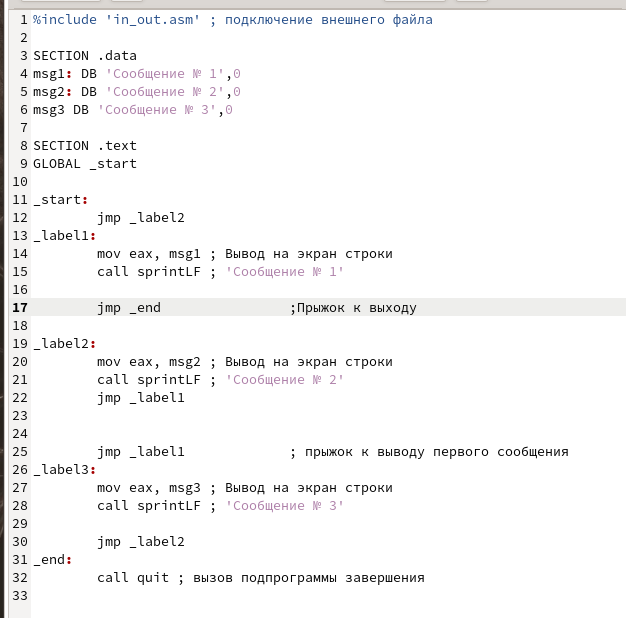


Рис. 4: Измененный lab8-1

Запустим програмуу и посмотрим на измененный резульат

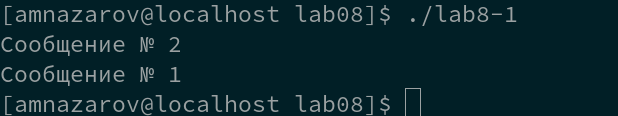


Рис. 5: Запуск lab8-1

## 2.4 Изменим код, что бы программа выводила 3, 2, 1

Нам достаточно перепрыгивать на label3 в начале, и в конце вывода “3” прыгать на label2, а дальше программа выполнгяется так как и раньше

Рис. 6: Измененный lab8-1

Рис. 6: Измененный lab8-1

Запустим программу и посмотрим на результат

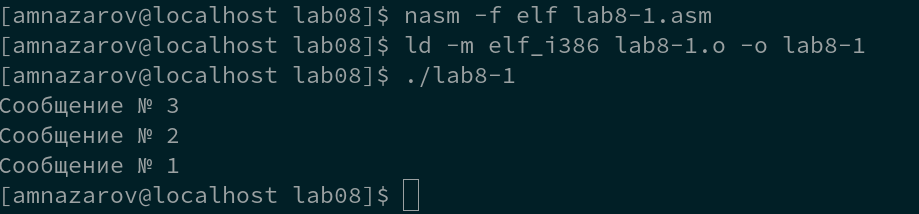


Рис. 7: Вывод измененного lab8-1

## 2.5 Создадим lab8-2 и откроем в Gedit

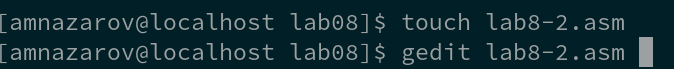


Рис. 8: Создание и отрытие lab8-2 в текстовом редакторе

Введем код в lab8-2

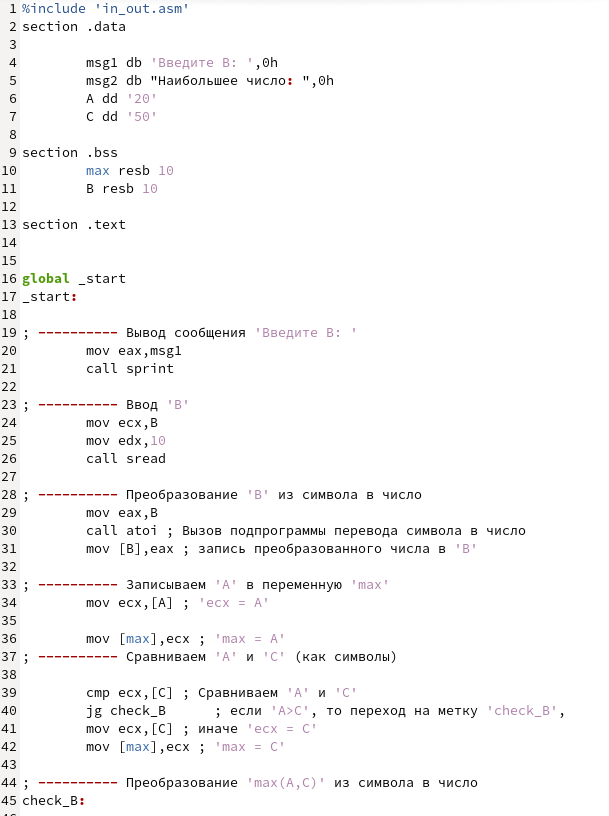


Рис. 9: Код lab8-2

Скомпилируем и исполним программу несколько раз

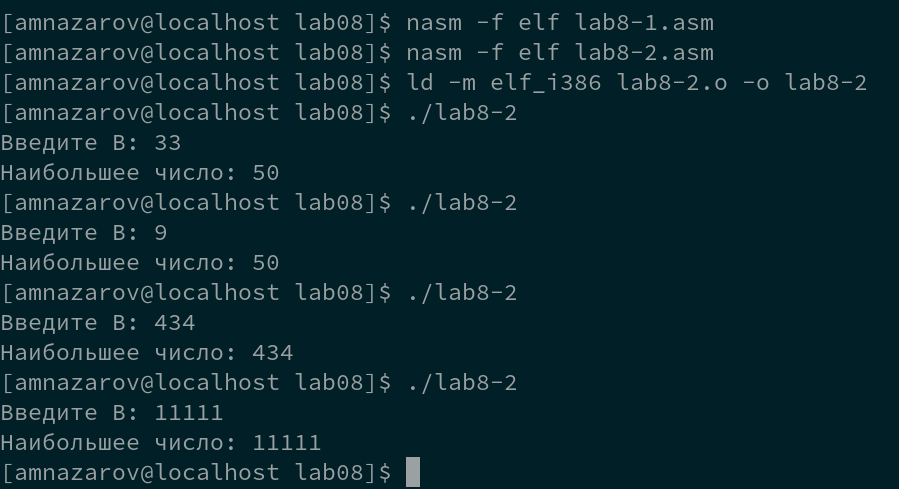


Рис. 10: Исполним программу несколько раз

## 2.6 Изучим листинг

Сгенерируем листинг

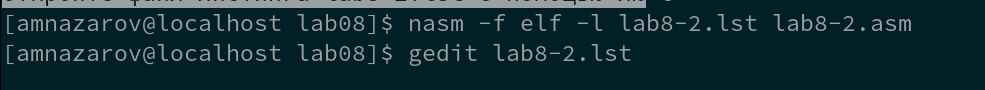


Рис. 11: Создание и открытие листинга в тестовом редакторе

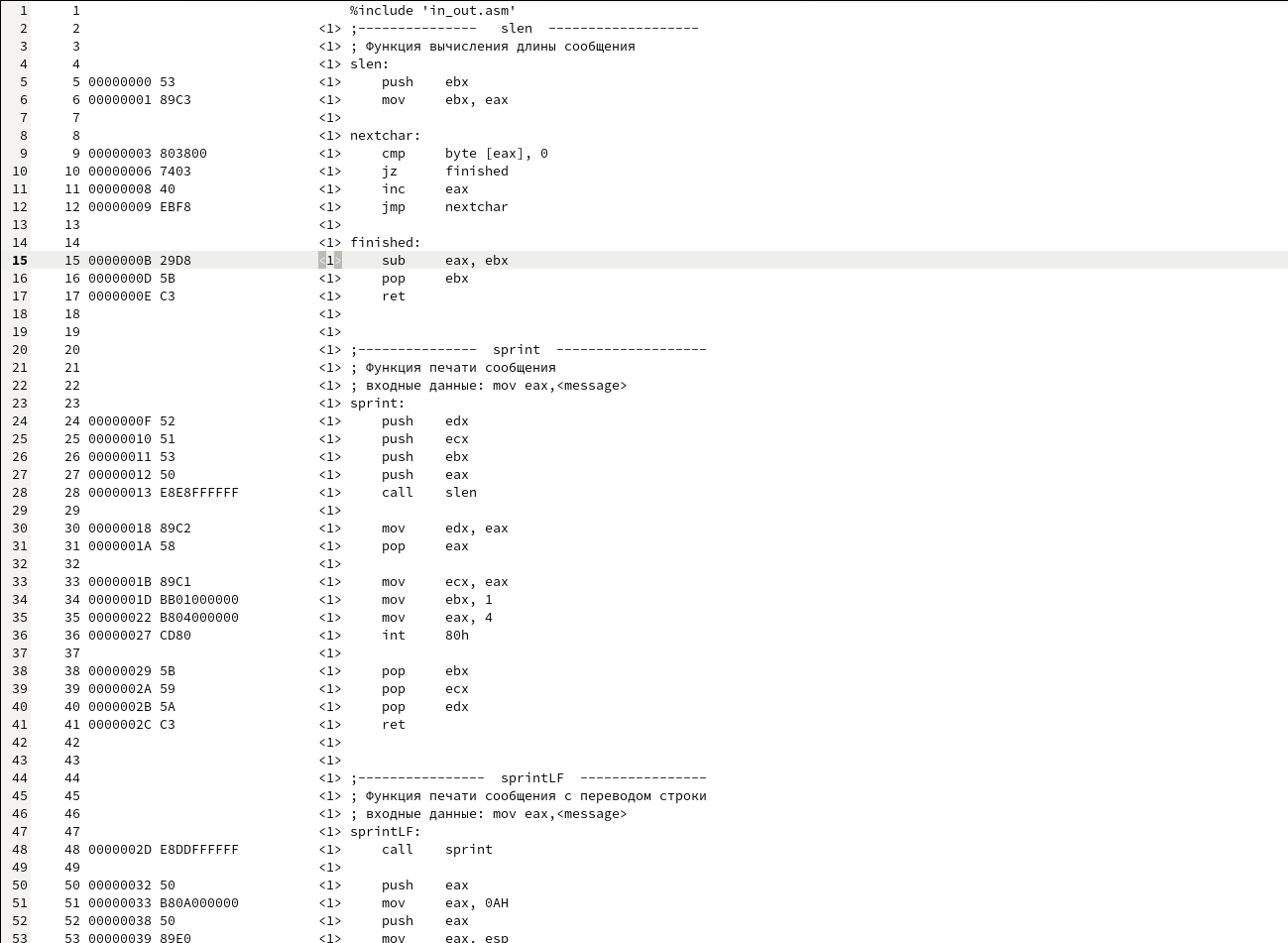


Рис. 12: Просмотр листинга в GEDIT

### 2.6.1 Описание трех строчек листинга

* На строке 25 видим номер строки (25), аддрес инструкции в памяти (00000010) и код комманды (51)
* Сравним строкит 5 и 16
* Аддрес инструкции на строке 5 - 00000000
* Аддрес инструкции на строке 16 000000D
* Код инструкции на строке 5 - 53
* Код инструкции на строке 16 5B
* Из этого можем сделать вывод, что код для ebx - 3, потому что 50 + 3 = 53, а 58+3 = 5B, так как код push 50, а pop 58

## 2.7 Изменим строчку в коде и создадим новый листинг

вместо cmp ecx, [C] оставим cmp ecx,

Рис. 13: Изменение lab8-2

Рис. 13: Изменение lab8-2

Сгенририуем листинг

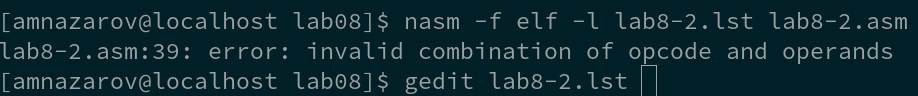


Рис. 14: Вывод при генерации листинга из кода с ошибкой

Откроем созданный файл листинга

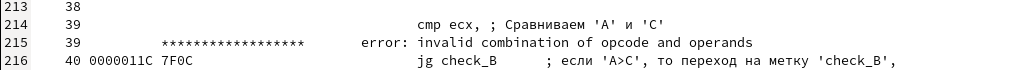


Рис. 15: Просмотр листинга

Видим, что выведенная при генерации ошибка находится в файле листинга, после измененной строчки

# 3 Задания для самостоятельной работы

Напишем код, который будет выводить наименьшее число из 84, 32, 77 (Так как мой номер 13)



Рис. 16: Код lab8-3.asm

Запустим программу и убедимся в правильности результата

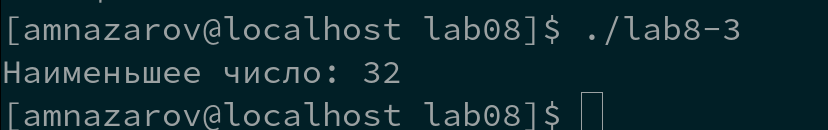


Рис. 17: Вывод lab8-3.asm

Напишем программу, которая будет вычислять функцию

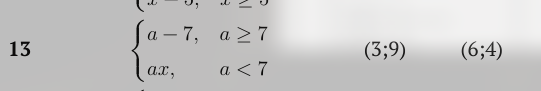


Рис. 18: Функция 13

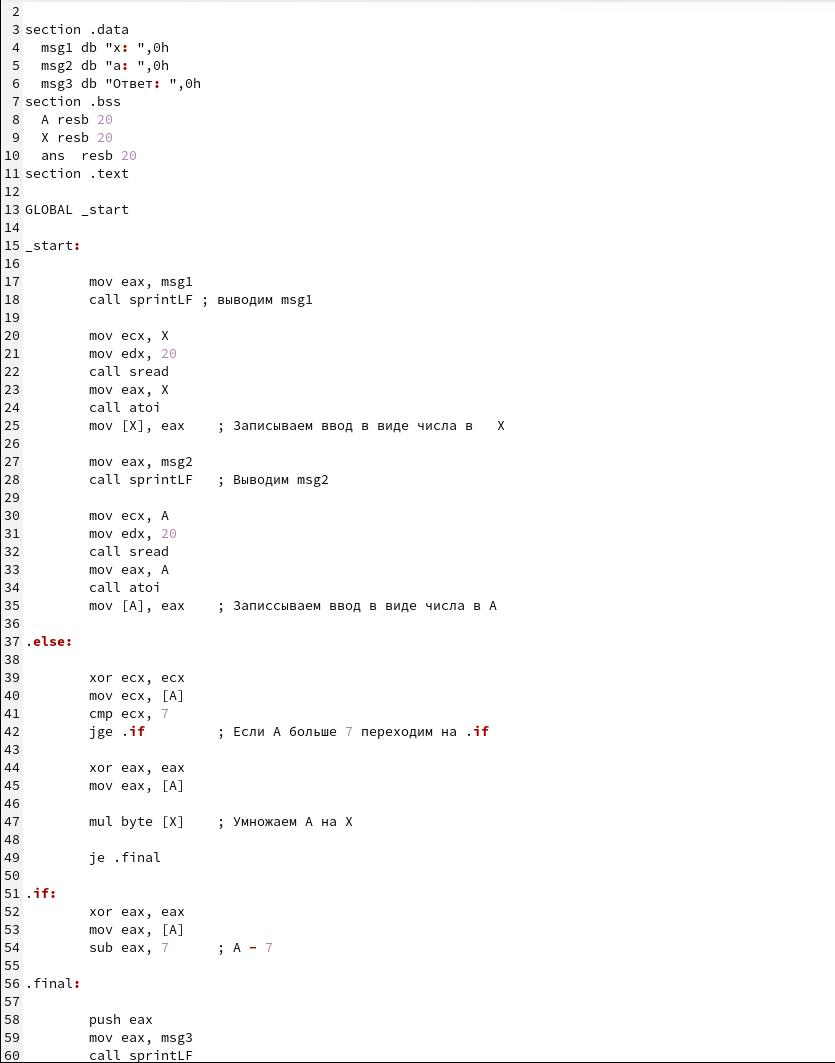


Рис. 19: Код lab8-4.asm

Запустим ее и проверим на данных значениях

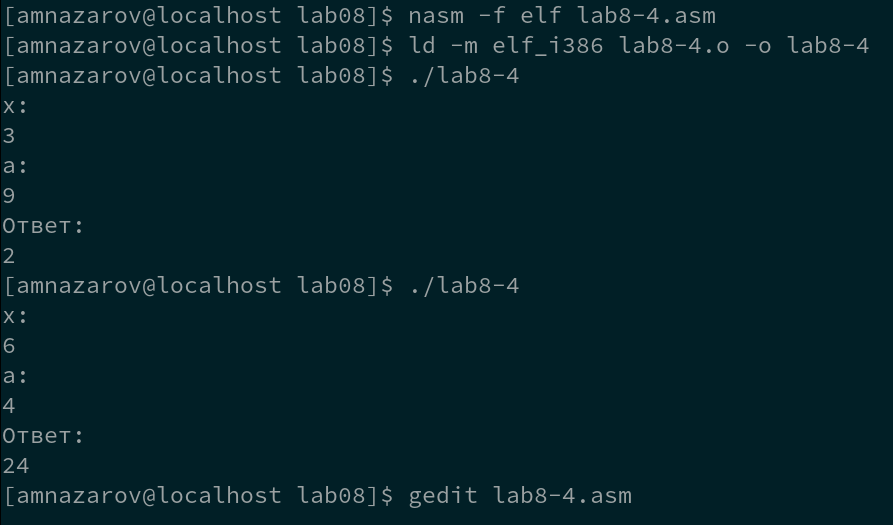


Рис. 20: Проверка lab8-4.asm

# 4 Выводы

Мы изучили комманды условного и безусловного переходов в языке асембела NASM, научились писать программы с использованием переходов и иознакомились с назначением и структурой файлов листинга

# Список литературы