Отчёт по лабораторной работе №8

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Кочина Дарья Сергеевна

Содержание

# 1 Цель работы

Целью данной лабораторной работы является изучение команд условного и безусловного переходов. А также приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

# 2 Задание

Изучить команды условного и безусловного переходов, приобрести практические навыки написания программ с использованием переходов и ознакомиться с назначением и структурой файла листинга.

# 3 Теоретическое введение

Для реализации ветвлений в ассемблере используются так называемые команды передачи управления или команды перехода. Можно выделить 2 типа переходов:

**условный переход** – выполнение или не выполнение перехода в определенную точку программы в зависимости от проверки условия.

**безусловный переход** – выполнение передачи управления в определенную точку программы без каких-либо условий.

Структура листинга:

*номер строки* — это номер строки файла листинга (нужно помнить, что номер строки в файле листинга может не соответствовать номеру строки в файле с исходным текстом программы);

*адрес* — это смещение машинного кода от начала текущего сегмента;

*машинный код* представляет собой ассемблированную исходную строку в виде шестнадцатеричной последовательности;

*исходный текст программы* — это просто строка исходной программы вместе с комментариями (некоторые строки на языке ассемблера, например, строки, содержащие только комментарии, не генерируют никакого машинного кода, и поля «смещение» и «исходный текст программы» в таких строках отсутствуют, однако номер строки им присваивается).

# 4 Выполнение лабораторной работы

1. Я создала каталог для программам лабораторной работы №8, перешла в него и создала файл lab8-1.asm. (рис. 1)

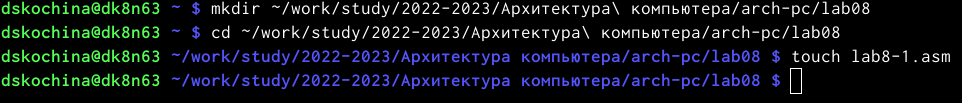


Рис. 1: Создание файла lab8-1.asm

1. Рассмотрела пример программы с использованием инструкции jmp. Для этого я ввела в файл lab8-1.asm текст программы из листинга. (рис. 2)

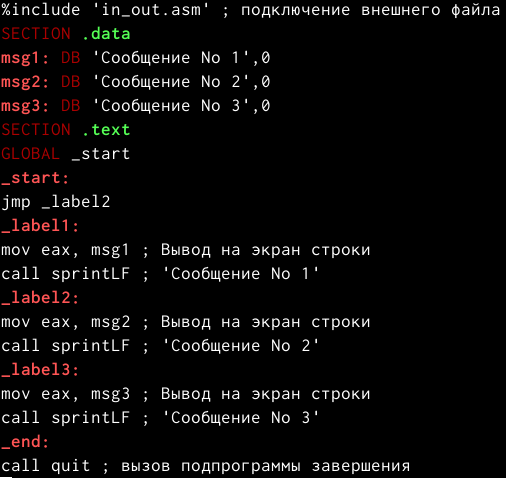


Рис. 2: Текст программы из листинга

1. Я создала исполняемый файл и запустила его. Результат работы данной программы был следующим. (рис. 3)

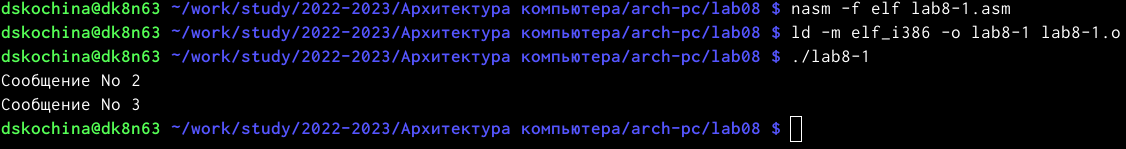


Рис. 3: Результат работы программы

1. Я изменила программу таким образом, чтобы она выводила сначала ‘Сообщение No 2’, потом ‘Сообщение No 1’ и завершала работу. Для этого в текст программы после вывода сообщения No 2 добавила инструкцию jmp с меткой \_label1 и после вывода сообщения No 1 добавила инструкцию jmp с меткой \_end. Я изменила текст программы в соответствии с листингом. (рис. 4, 5)

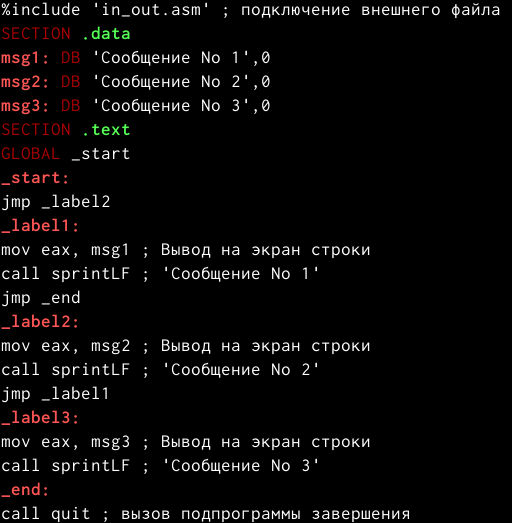


Рис. 4: Текст программы из листинга

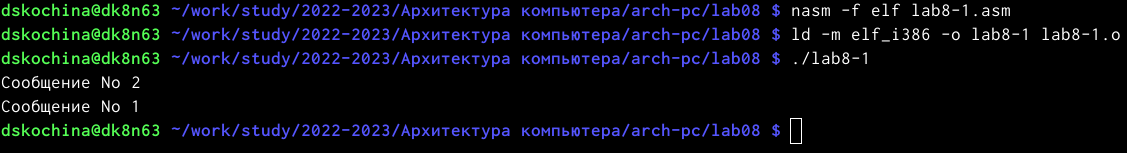


Рис. 5: Результат работы программы

1. Я изменила текст программы, изменив инструкции jmp, чтобы вывод программы был следующим. (рис. 6, 7)

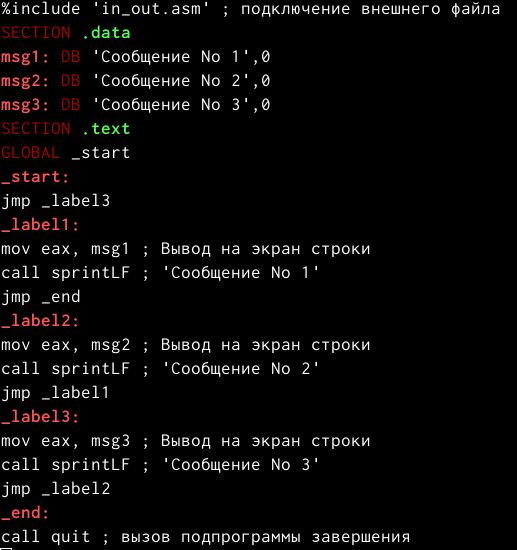


Рис. 6: Текст программы из листинга

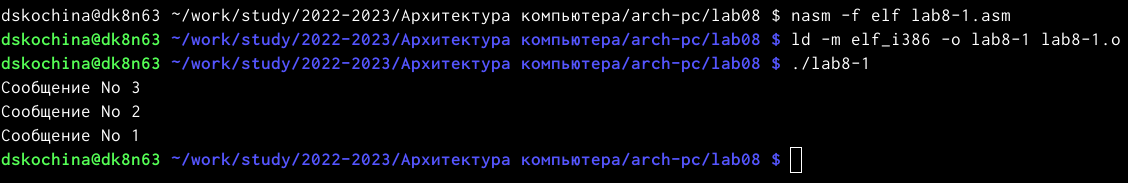


Рис. 7: Результат работы программы

1. Я создала файл lab8-2.asm. Внимательно изучила текст программы из листинга (программа, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных: A, B и C). Я ввела его в lab8-2.asm. (рис. 8, 9, 10)

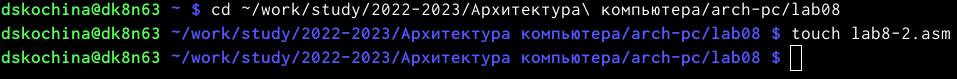


Рис. 8: Создание файла lab8-2.asm

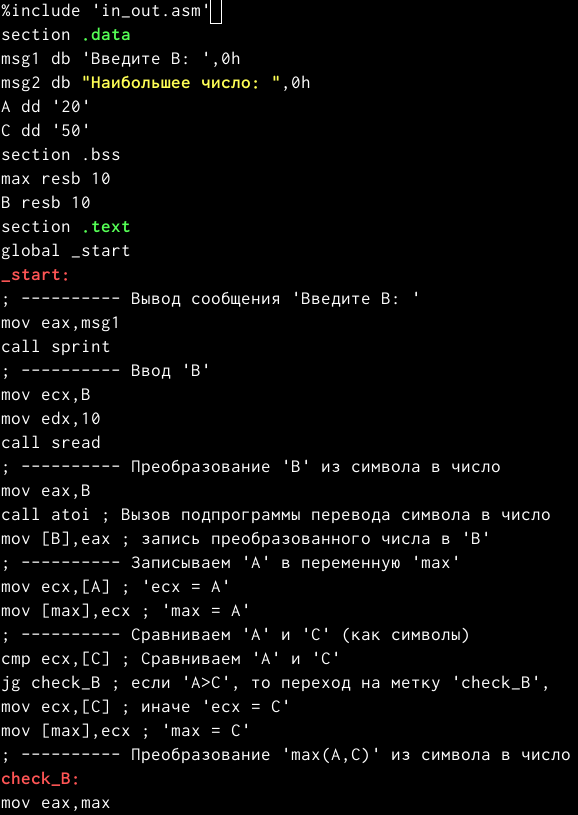


Рис. 9: Текст программы из листинга

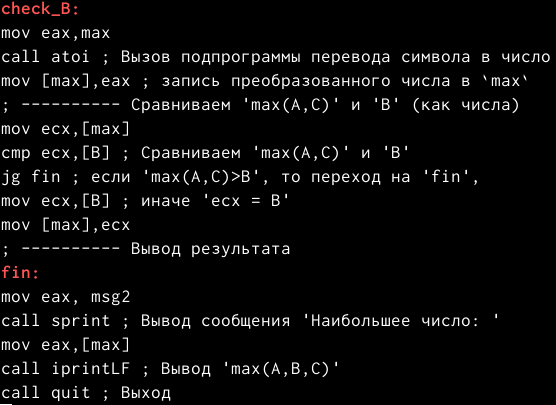


Рис. 10: Текст программы из листинга

1. Я создала исполняемый файл и проверила его работу для разных значений B. (рис. 11)

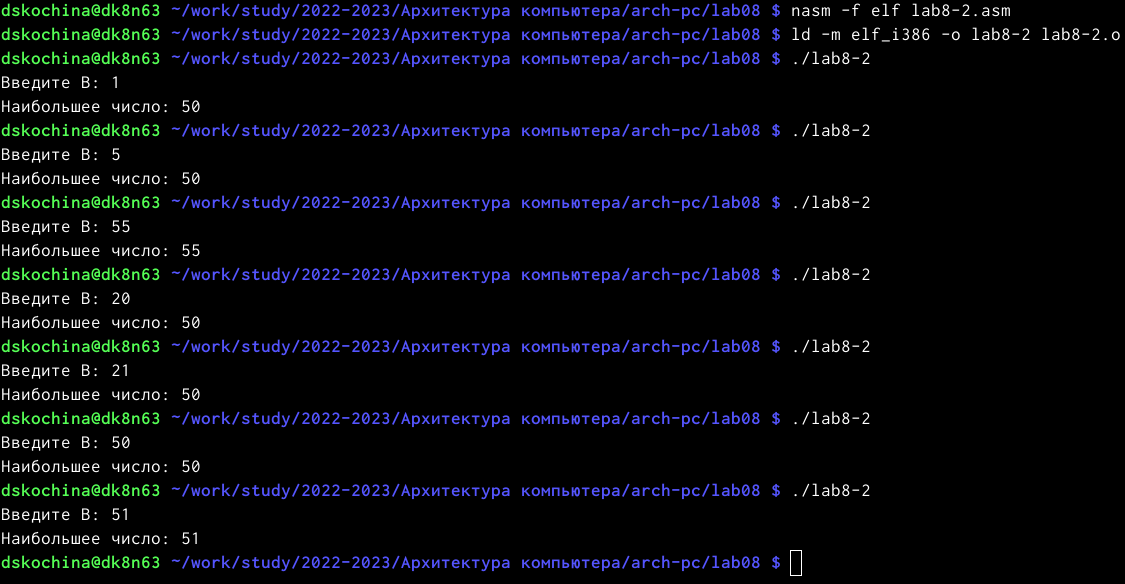


Рис. 11: Результат работы программы

1. Я создала файл листинга для программы из файла lab8-2.asm, указав ключ -l и задав имя файла листинга в командной строке. Затем я открыла файл листинга lab8-2.lst с помощью текстового редактора mcedit. Внимательно ознакомилась с его форматом и содержимым. (рис. 12, 13)

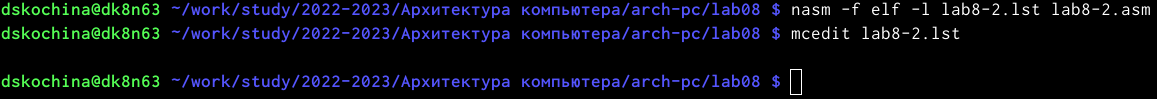


Рис. 12: Создание файла листинга для программы

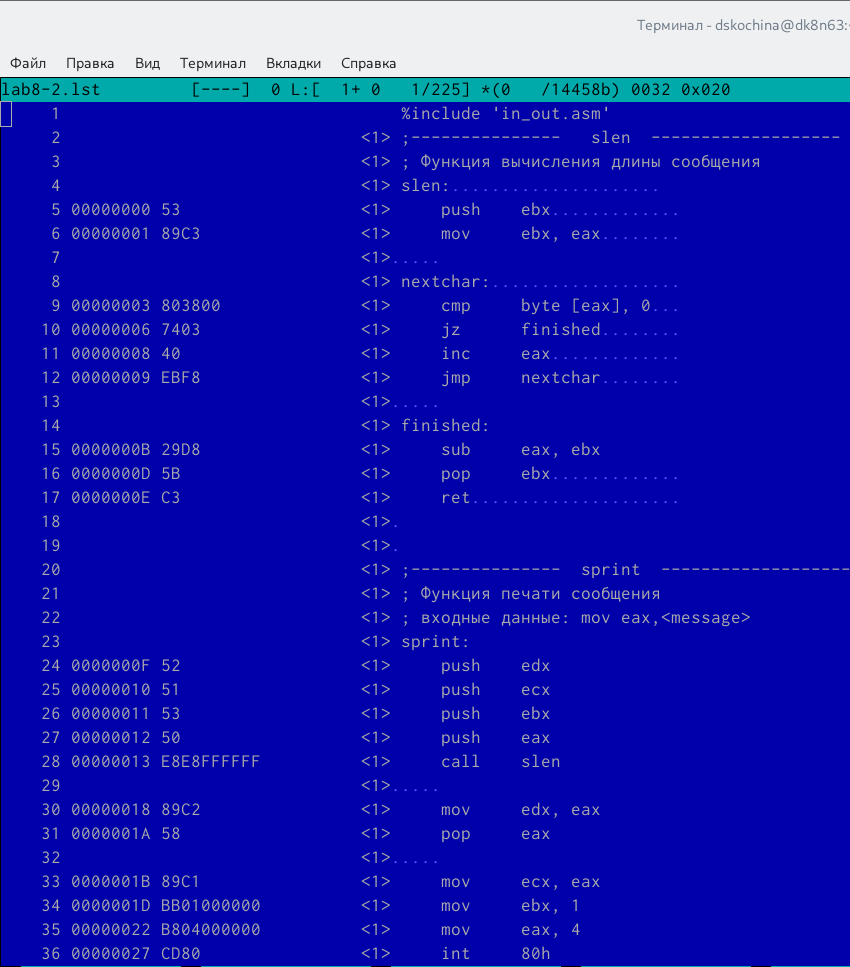


Рис. 13: Открытие файла в редакторе

1. Я открыла файл с программой lab8-2.asm и в строке mov eax,max я убрала max и попробовала создать файл. Затем я выполнила трансляцию с получением файла листинга. Выдало ошибку, поскольку для программы нужно два операнда. (рис. 14, 15)

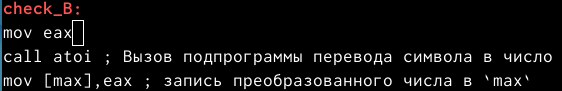


Рис. 14: Удаление одного операнда

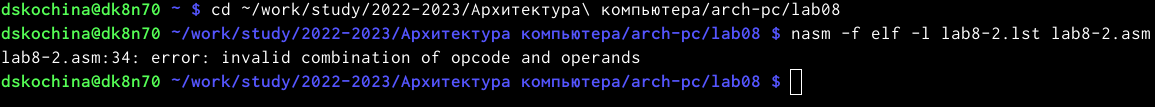


Рис. 15: Создание файла без одного операнда

1. В файле листинга можно увидеть, где именно ошибка и с чем она связана. (рис. 16)

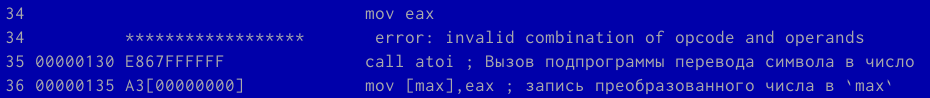


Рис. 16: Файл листинга без одного операнда

1. Эта строка находится на 35 месте, её адрес 00000130, машинный код - E867FFFFFF, исходный текст программы - call atoi. Он означает, что символ, лежащий в строке выше, переводится в число. (рис. 17)

Рис. 17: Объяснение первой строки

Рис. 17: Объяснение первой строки

1. Эта строка находится на 47 месте, её адрес 0000012E, машинный код - A1[00000000], исходный текст программы - mov eax,[max]. Он означает, что число, хранящееся в переменной max, записывается в eax. (рис. 18)

Рис. 18: Объяснение второй строки

Рис. 18: Объяснение второй строки

1. Эта строка находится на 21 месте, её адрес 00000101, машинный код - B8[0A000000], исходный текст программы - mov eax,B. Он означает, что в регистр eax мы вносим значения переменной B. (рис. 19)

Рис. 19: Объяснение третьей строки

Рис. 19: Объяснение третьей строки

**Самостоятельная работа**

1. Я написала программу нахождения наименьшей из 3 целочисленных переменных a,b,с. Значения переменных я выбрала из таблицы (81,22,72) в соответсвии с вариантом, полученным при выполнении лабораторной работы №7 (вариант 14). Я создала исполняемый файл и проверила его работу. (рис. 20, 21, 22)

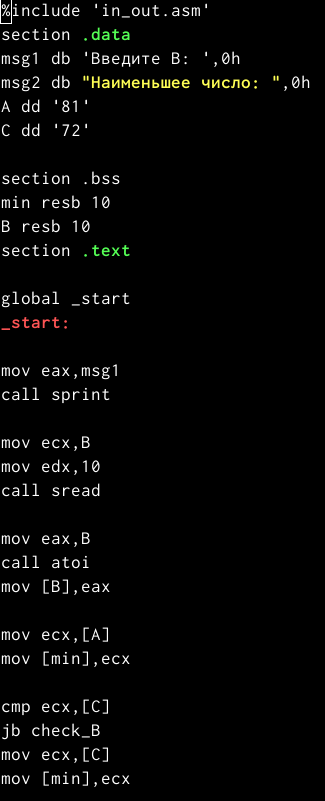


Рис. 20: Текст программы в файле lab8-3.asm

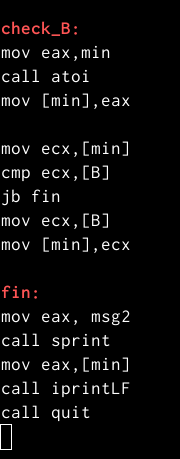


Рис. 21: Результат работы программы

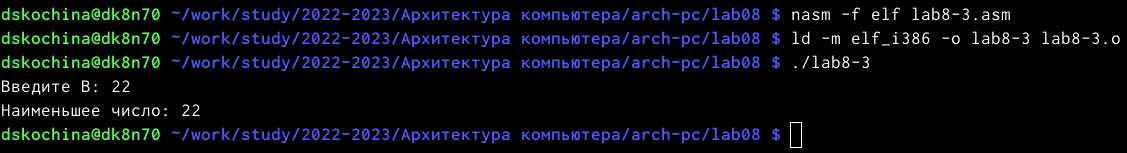


Рис. 22: Результат работы программы

1. Я написала программу, которая для введённых с клавиатуры значений x и a вычисляет значение заданной функции f(x) и выводит результат вычислений. Вид функции f(x) я выбрала из таблицы вариантов заданий в соответствии с вариантом, полученным при выполнении работы №7 (вариант 14). Я создала исполняемый файл и проверила его работу для значений x и a. (рис. 23, 24, 25)

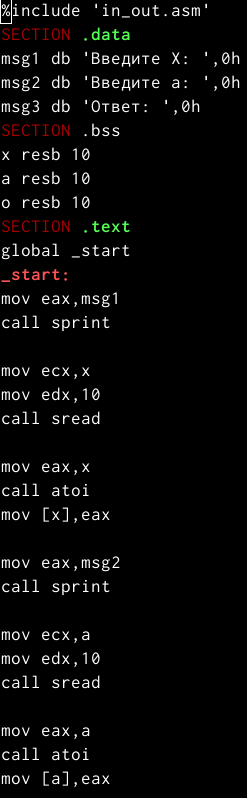


Рис. 23: Текст программы в файле lab8-4.asm

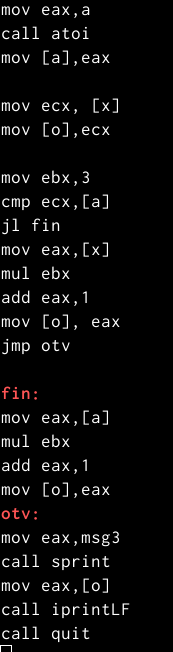


Рис. 24: Результат работы программы

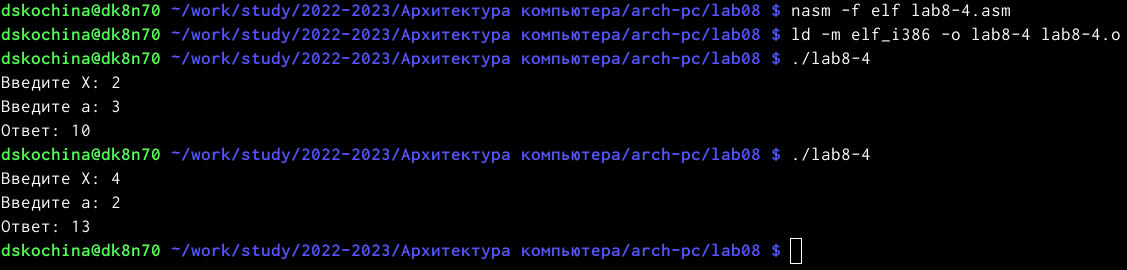


Рис. 25: Результат работы программы

# 5 Выводы

В ходе выполнения данной лабораторной работы я изучила команды условного и безусловного переходов. А также приобрела навыки написания программ с использованием переходов и ознакомилась с назначением и структурой файла листинга.