Отчёт по лабораторной работе 7

Архитектура компьютеров и операционные системы

Горелашвили Лия Михайловна НКАбд-02-23

Содержание

# 1 Цель работы

Целью работы является изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

# 2 Задание

1. Изучение команд условного и бкзусловного перехода
2. Изучение файла листинга
3. Выполнение заданий, рассмотрение примеров
4. Выполнение заданий для самостоятельной работы

# 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.

Безусловный переход выполняется инструкцией jmp (от англ. jump – прыжок), которая включает в себя адрес перехода, куда следует передать управление

Инструкция cmp является одной из инструкций, которая позволяет сравнить операнды и выставляет флаги в зависимости от результата сравнения. Инструкция cmp является командой сравнения двух операндов и имеет такой же формат, как и команда вычитания

Команда условного перехода имеет вид

j<мнемоника перехода> label

Мнемоника перехода связана со значением анализируемых флагов или со способом формирования этих флагов.

# 4 Выполнение лабораторной работы

## 4.1 Реализация переходов в NASM

Создала каталог для программ лабораторной работы No7 и файл с названием “lab7-1.asm”. Инструкция jmp в NASM используется для безусловных переходов.

Давайте рассмотрим пример программы с использованием jmp. Написала текст программы из листинга 7.1 в файле “lab7-1.asm”.

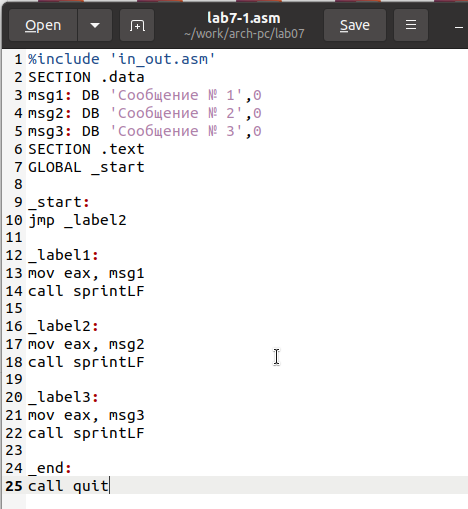


Figure 1: Редактирование файла lab7-1.asm

Затем создала исполняемый файл и запустила его.

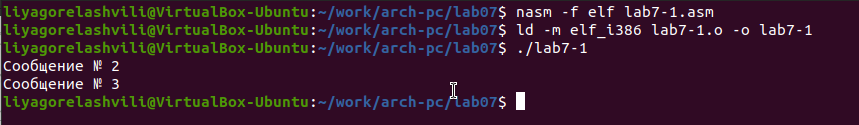


Figure 2: Проверка кода lab7-1.asm

Инструкция jmp позволяет осуществлять переходы не только вперед, но и назад. Изменила программу так, чтобы сначала выводилось “Сообщение No2”, потом “Сообщение No1”, а затем происходил выход. Для этого после вывода “Сообщения No2” добавила инструкцию jmp с меткой “\_label1” (переход к выводу “Сообщения No1”). А после вывода “Сообщения No1” добавила инструкцию jmp с меткой “\_end” (переход к инструкции call quit). Изменила текст программы в соответствии с листингом 7.2.

Изменила текст программы в соответствии с листингом 7.2.

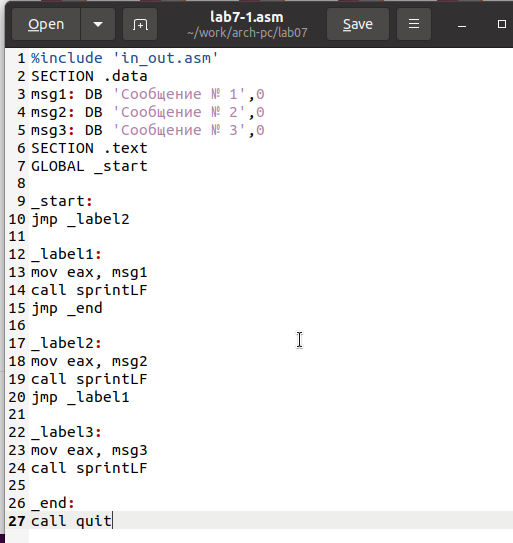


Figure 3: Редактирование файла lab7-1.asm

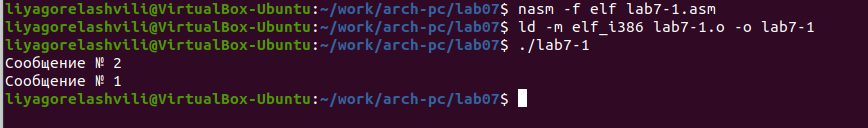


Figure 4: Проверка кода lab7-1.asm

Изменила текст программы, изменив инструкции jmp, чтобы вывод программы был следующим:

Сообщение № 3  
Сообщение № 2  
Сообщение № 1

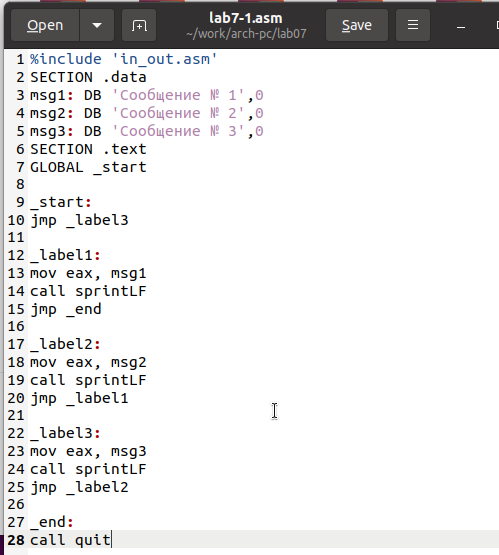


Figure 5: Редактирование файла lab7-1.asm

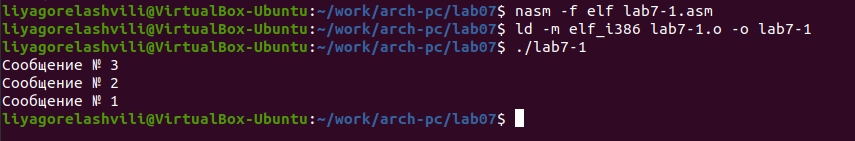


Figure 6: Проверка кода lab7-1.asm

Использование инструкции jmp приводит к переходу в любом случае. Однако, часто при написании программ необходимо использовать условные переходы, то есть переход должен происходить, если выполнено какое-либо условие.

Я рассмотрела программу, которая определяет и выводит наибольшее из трех чисел: A, B и C. Значения для A и C задаются в коде, а значение B вводится с клавиатуры.

Создала исполняемый файл и проверила его работу для разных значений B.

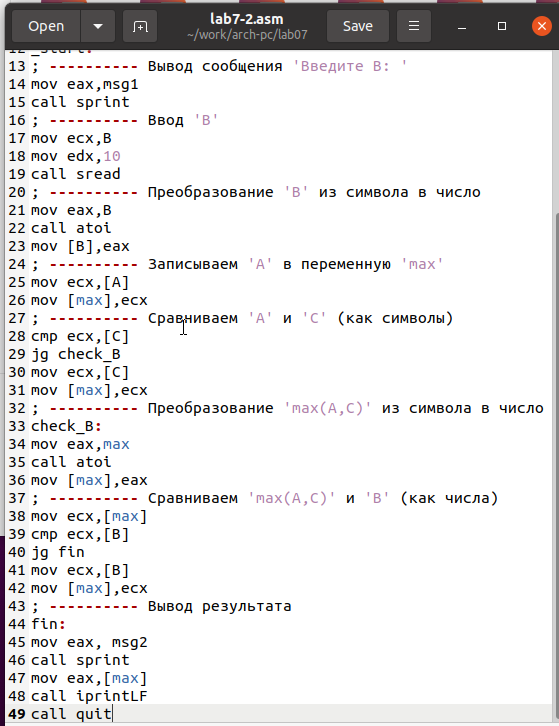


Figure 7: Редактирование файла lab7-2.asm

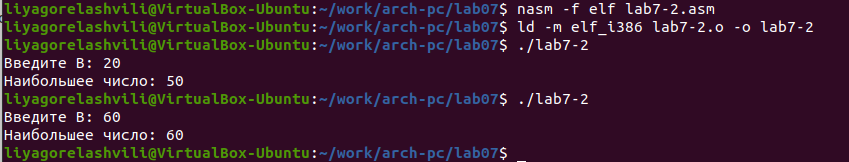


Figure 8: Проверка кода lab7-2.asm

## 4.2 Изучение структуры файлы листинга

Обычно nasm создаёт только объектный файл после ассемблирования. Чтобы получить файл листинга, нужно указать ключ -l и задать имя файла листинга в командной строке.

Я создала файл листинга для программы из lab7-2.asm.

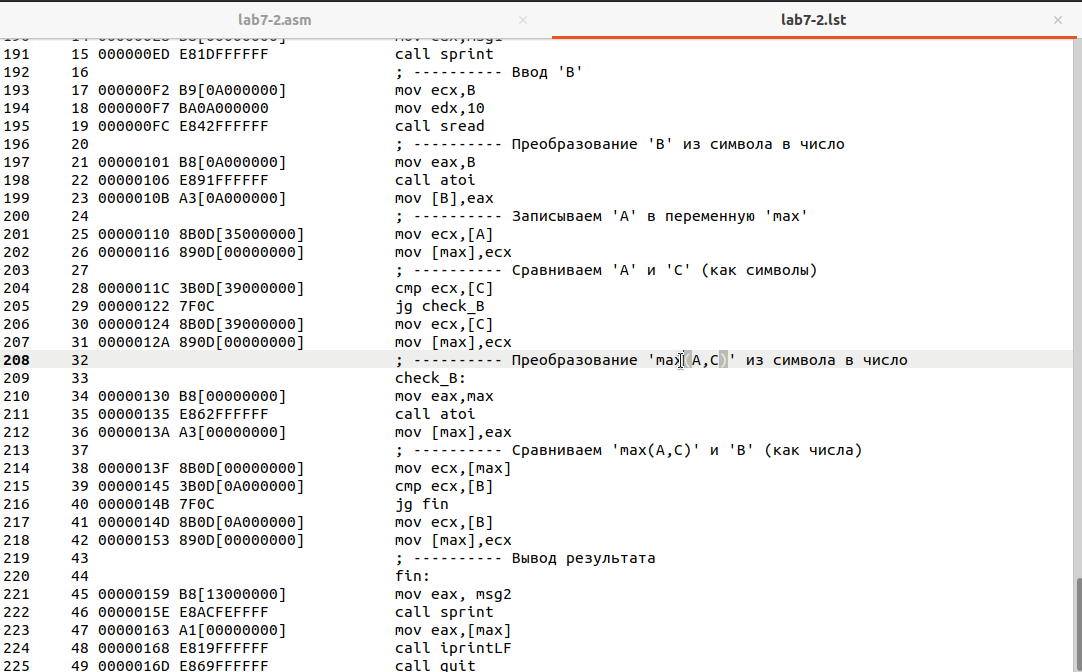


Figure 9: Файл листинга lab7-2

Внимательно ознакомилась с его форматом и содержимым. Подробно объясню содержимое трёх строк этого листинга.

*строка 209*

* 34 - номер строки в подпрограмме
* 00000130 - адрес
* B8[00000000] - машинный код
* mov eax,max - код программы - копирует max в eax

*строка 210*

* 35 - номер строки в подпрограмме
* 00000135 - адрес
* E862FFFFFF - машинный код
* call atoi - код программы - вызов подпрограммы atoi

*строка 211*

* 36 - номер строки в подпрограмме
* 0000013A - адрес
* A3[00000000] - машинный код
* mov [max],eax - код программы - копирует eax в max

Открыла файл с программой lab7-2.asm и в инструкции с двумя операндами удалила один операнд. Выполнила трансляцию с получением файла листинга.

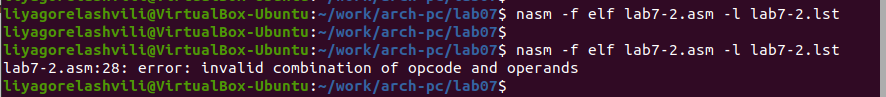


Figure 10: Ошибка трансляции lab7-2

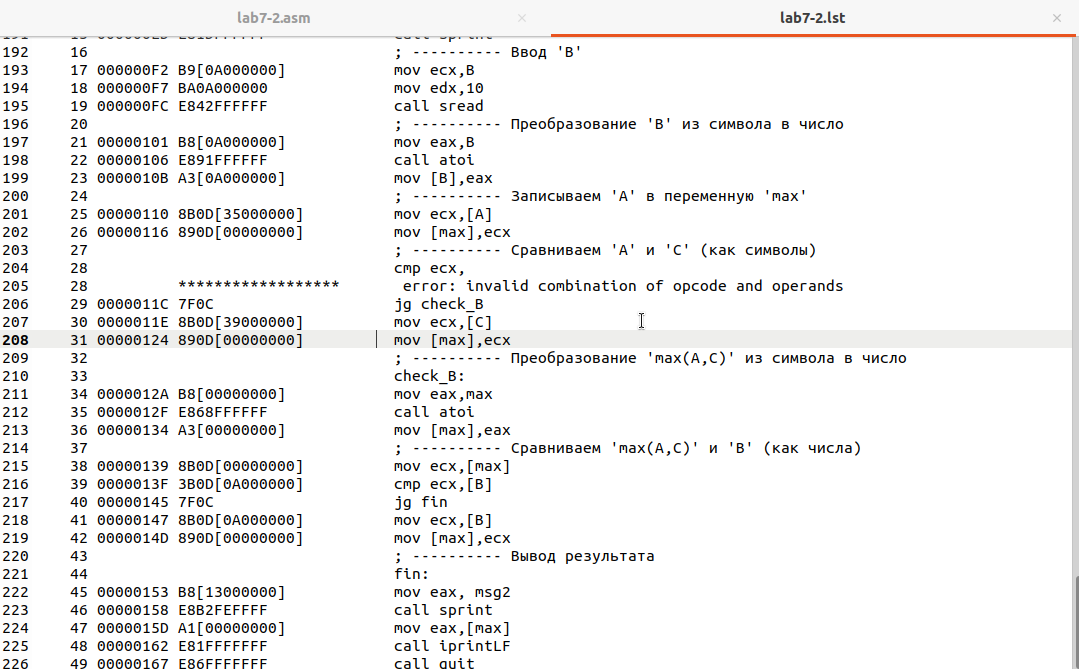


Figure 11: Файл листинга с ошибкой lab7-2

Объектный файл не смог создаться из-за ошибки. Но получился листинг, где выделено место ошибки.

## 4.3 Выполнение заданий для самостоятельной работы

Напишите программу нахождения наименьшей из 3 целочисленных переменных a,b и c.  Значения переменных выбрать из табл. 7.5 в соответствии с вариантом, полученным при выполнении лабораторной работы № 6. Создайте исполняемый файл и проверьте его работу.

Мой вариант 8 - числа: 52,33,40

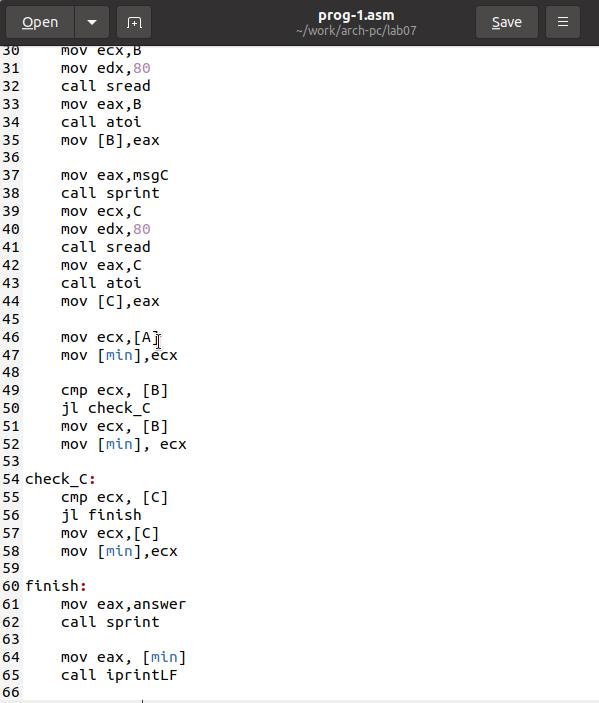


Figure 12: Редактирование файла prog-1.asm

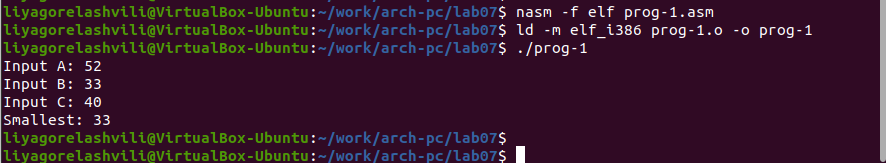


Figure 13: Проверка кода prog-1.asm

Напишите программу, которая для введенных с клавиатуры значений x и a вычисляет значение заданной функции f(x) и выводит результат вычислений. Вид функции f(x) выбрать из таблицы 7.6 вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы № 7. Создайте исполняемый файл и проверьте его работу для значений X и a из 7.6. (рис. [[14](#fig:014)]) (рис. [[15](#fig:015)])

Мой вариант 8

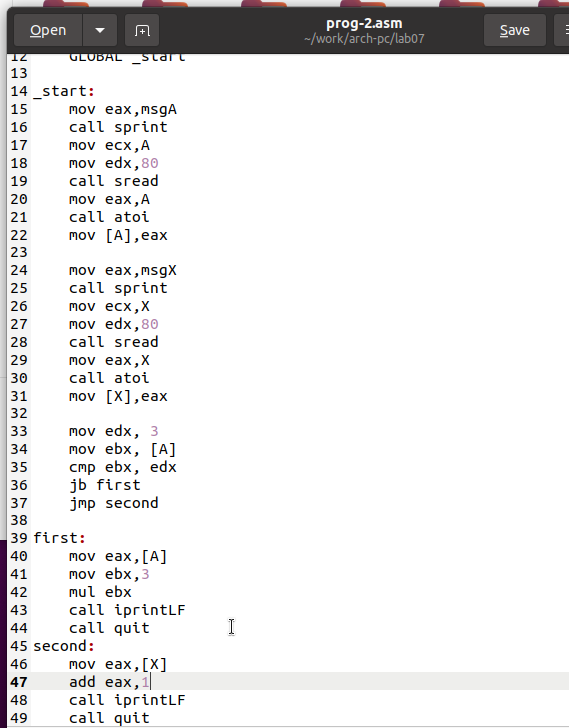


Figure 14: Редактирование файла prog-2.asm

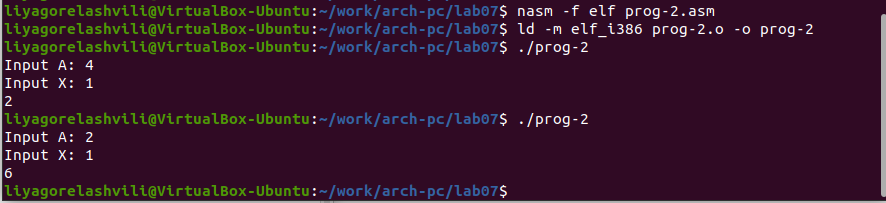


Figure 15: Проверка кода prog-2.asm

# 5 Выводы

Изучили команды условного и безусловного переходов, познакомились с фалом листинга.