Отчёт по лабораторной работе 7

Архитектура компьютера

Амарбаяр Чинхусэл

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 Выполнение лабораторной работы

1. Создал каталог для программам лабораторной работы № 7 и файл lab7-1.asm
2. Инструкция jmp в NASM используется для реализации безусловных переходов. Рассмотрим пример программы с использованием инструкции jmp.

Написал в файл lab7-1.asm текст программы из листинга 7.1.

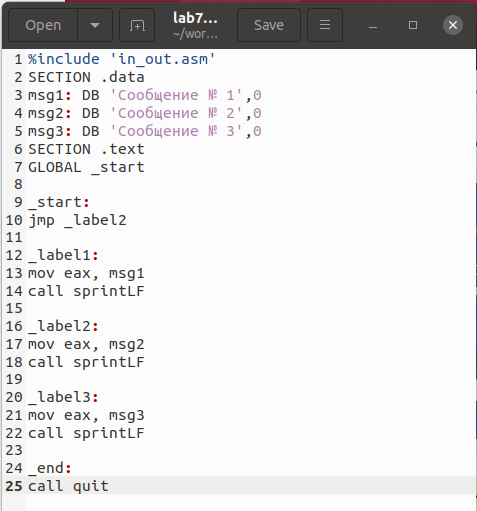


Figure 1: Программа lab7-1.asm

Создал исполняемый файл и запустил его.

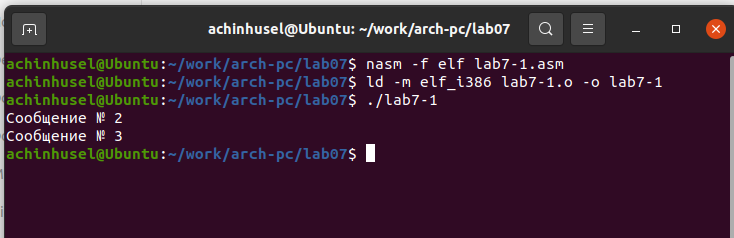


Figure 2: Запуск программы lab7-1.asm

Инструкция jmp позволяет осуществлять переходы не только вперед но и назад. Изменим программу таким образом, чтобы она выводила сначала ‘Сообщение № 2’, потом ‘Сообщение № 1’ и завершала работу. Для этого в текст программы после вывода сообщения № 2 добавим инструкцию jmp с меткой \_label1 (т.е. переход к инструкциям вывода сообщения № 1) и после вывода сообщения № 1 добавим инструкцию jmp с меткой \_end (т.е. переход к инструкции call quit).

Изменил текст программы в соответствии с листингом 7.2.

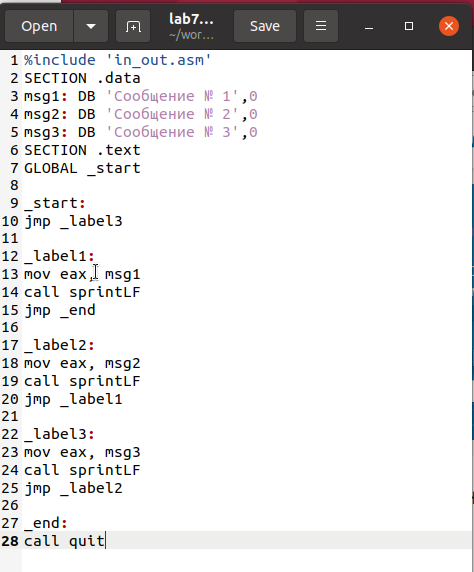


Figure 3: Программа lab7-1.asm

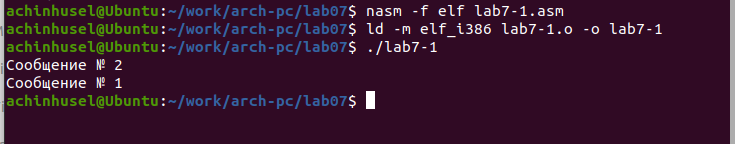


Figure 4: Запуск программы lab7-1.asm

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

Сообщение № 3  
Сообщение № 2  
Сообщение № 1

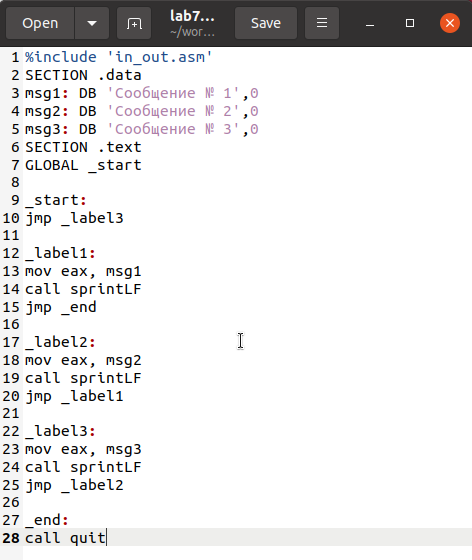


Figure 5: Программа lab7-1.asm

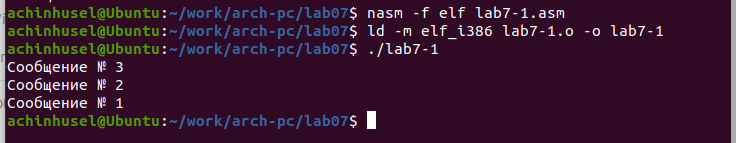


Figure 6: Запуск программы lab7-1.asm

1. Использование инструкции jmp приводит к переходу в любом случае. Однако, часто при написании программ необходимо использовать условные переходы, т.е. переход должен происходить если выполнено какое-либо условие. В качестве примера рассмотрим программу, которая определяет и выводит на экран наибольшую из 3 целочисленных переменных: A,B и C. Значения для A и C задаются в программе, значение B вводиться с клавиатуры.

Создал исполняемый файл и проверил его работу для разных значений B.

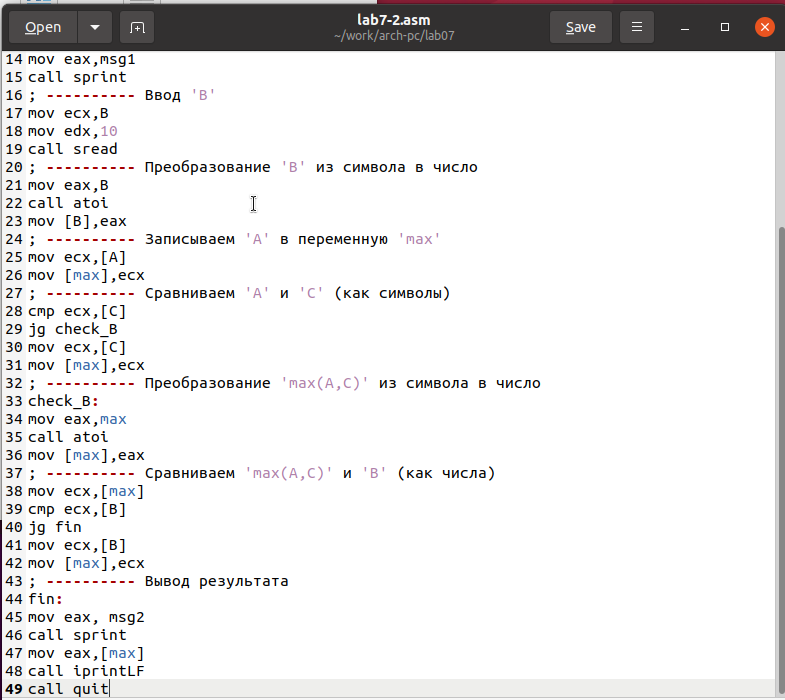


Figure 7: Программа lab7-2.asm

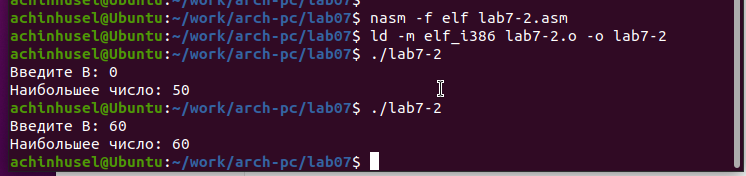


Figure 8: Запуск программы lab7-2.asm

1. Обычно nasm создаёт в результате ассемблирования только объектный файл. Получить файл листинга можно, указав ключ -l и задав имя файла листинга в командной строке.

Создал файл листинга для программы из файла lab7-2.asm

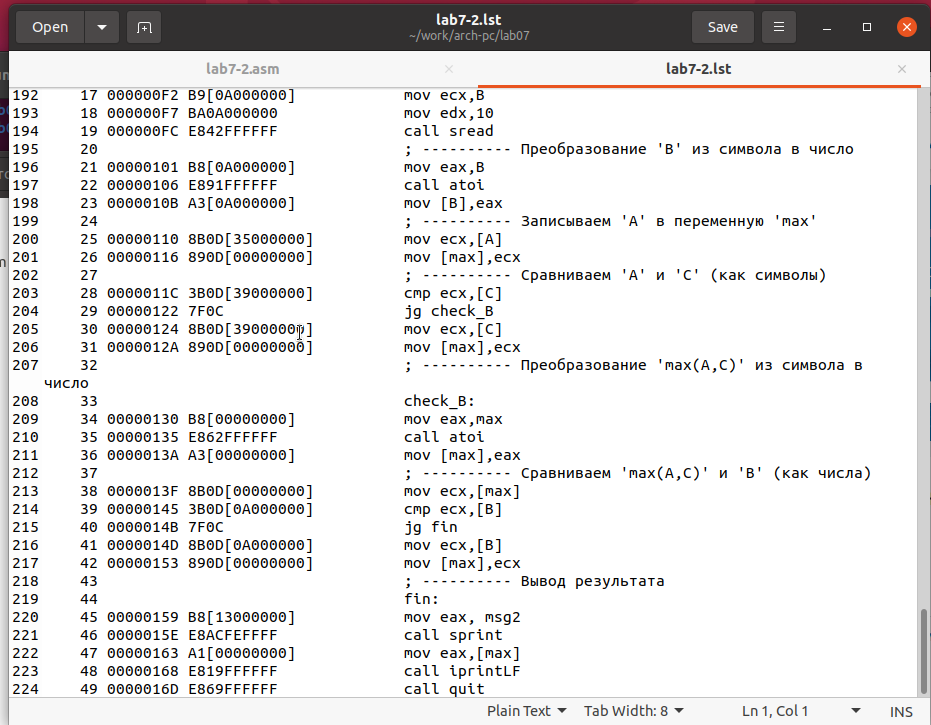


Figure 9: Файл листинга lab7-2

Внимательно ознакомился с его форматом и содержимым. Подробно объясню содержимое трёх строк файла листинга по выбору.

\*строка 209

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

\*строка 210

- 35 - номер строки  
  
- 00000135 - адрес  
  
- E862FFFFFF - машинный код  
  
- call atoi - код программы

\*строка 211

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

Открыл файл с программой lab7-2.asm и в инструкции с двумя операндами удалил один операнд. Выполнил трансляцию с получением файла листинга.

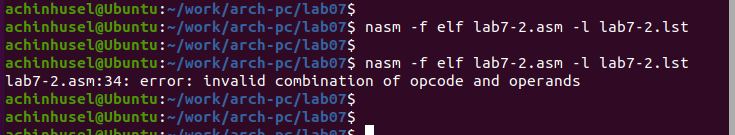


Figure 10: Ошибка трансляции lab7-2

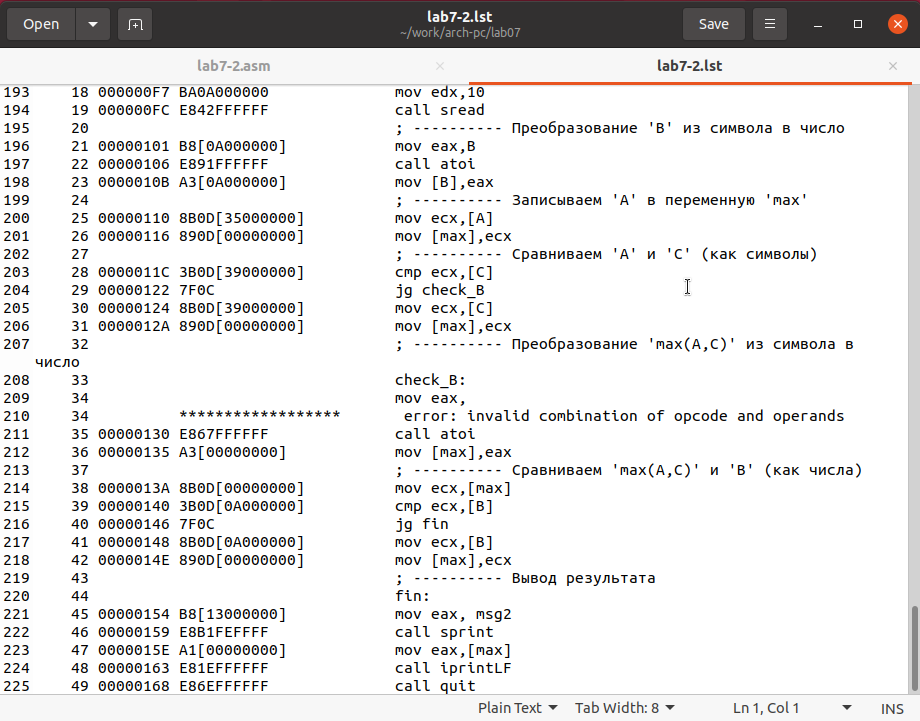


Figure 11: Файл листинга с ошибкой lab7-2

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

1. 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

для варианта 20 - 92, 2, 61

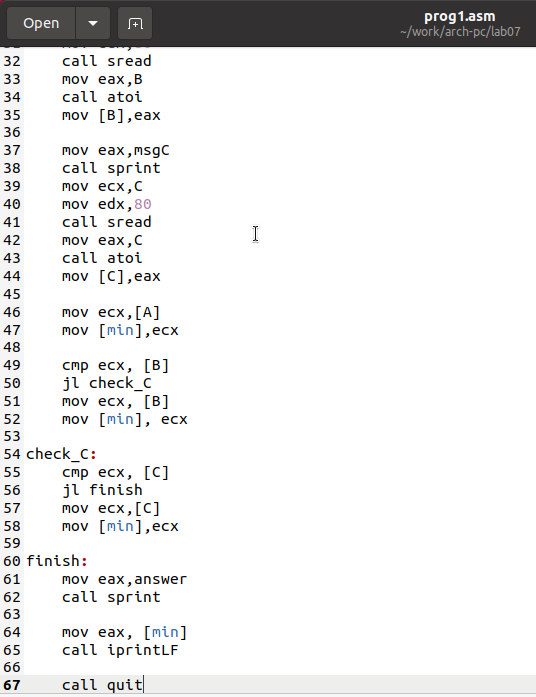


Figure 12: Программа prog1.asm

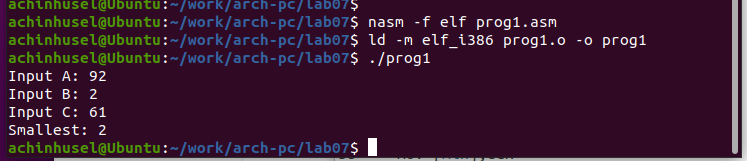


Figure 13: Запуск программы prog1.asm

1. 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.

для варианта 20

[Программа prog2.asm]](image/14.png){ #fig:014 width=70%, height=70% }

Подставляем и

[Запуск программы prog2.asm]](image/15.png){ #fig:015 width=70%, height=70% }

# 3 Выводы

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