Лабораторная работа №7

Команды безусловного и условного переходов в Nasm. Программирование ветвлений.

Казначеев Сергей Ильич

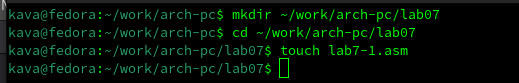
Содержание

# 1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

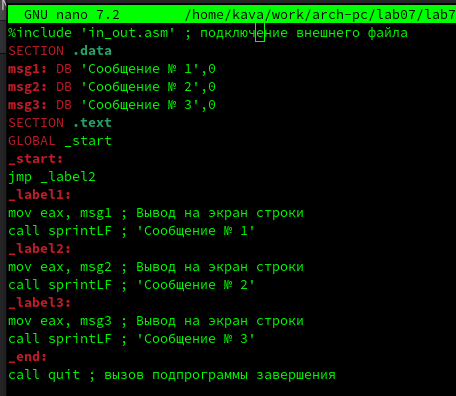
# 2 Выполнение лабораторной работы

Создаем папку lab07 и файл lab7-1.asm.



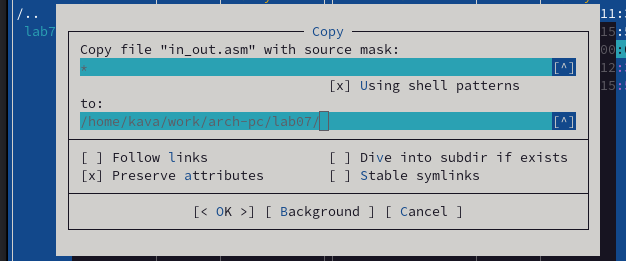
01

После чего, запускаем Midnight commander для удобства и вставляем код в файл lab7-1.asm из файла листинга.



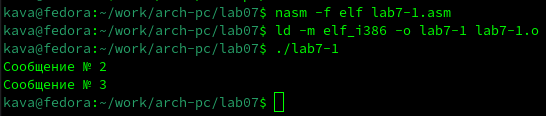
02

Теперь скопируем файл in\_out.asm из рабочей директории прошлой лабораторной работы.



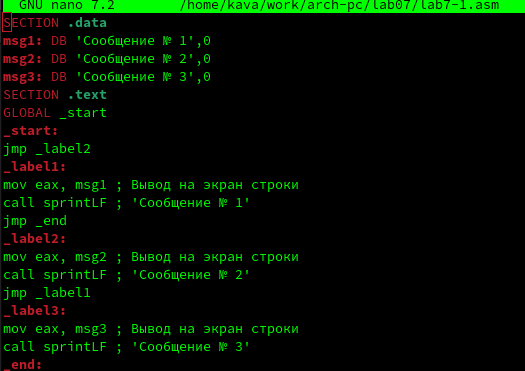
03

Теперь собираем программу и запускаем ее.



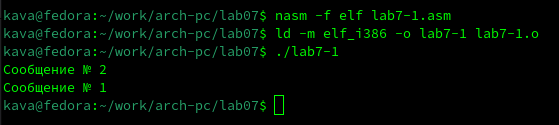
04

После чего, мы меням файл согласно листингу 7.2



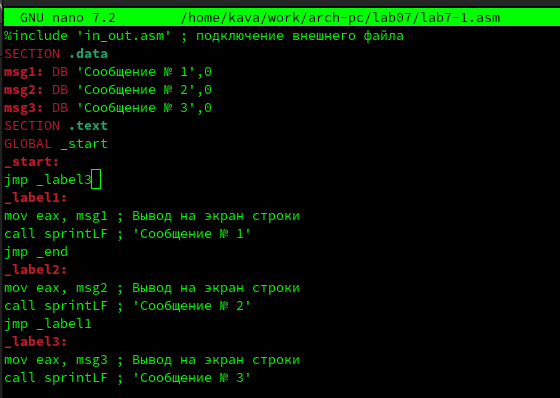
05

Снова соберём программу и запустим ёё.



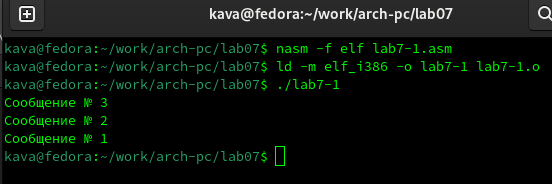
06

Теперь сделаем так, чтобы код выводил сообщения в обратном порядке (от 3 сообщения к первому). Для этого внесём в код следующие изменения.



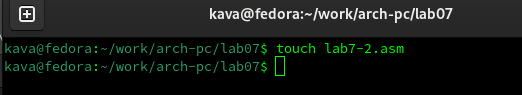
07

Снова соберём программу и запустим ёё.



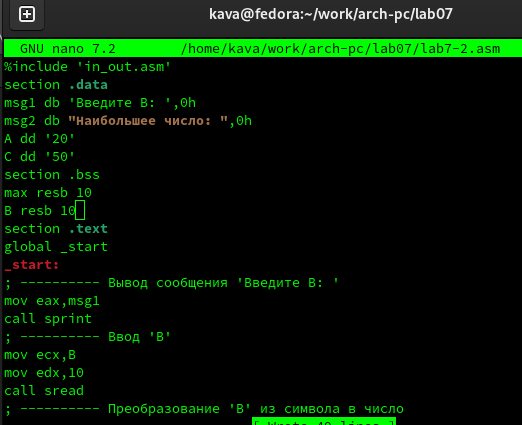
08

Теперь создаем файл lab7-2.asm.



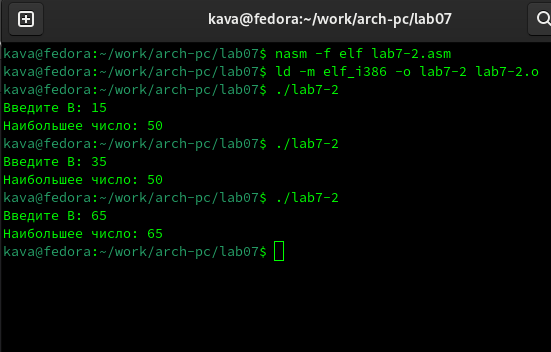
09

Запишем в него код из листинга 7.3 в файл lab7-2.asm.



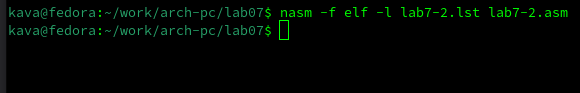
10

И запустим его, предворительно собрав.



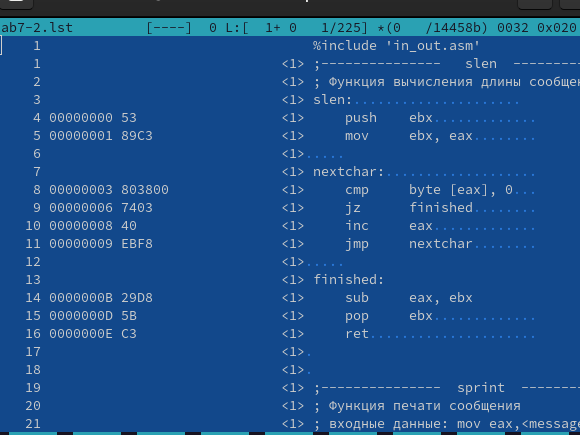
11

Теперь попробуем создать файл листинга при сборке файла lab7-2.asm



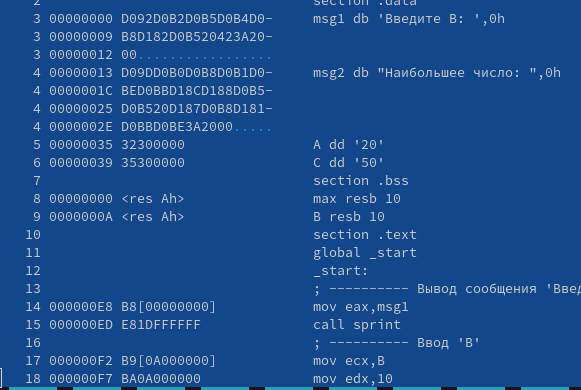
12

Теперь посмотрим, как выглядит файл листинга изнутри. Для этого откроем его в mcedit.



13

Вот наша программа.



14

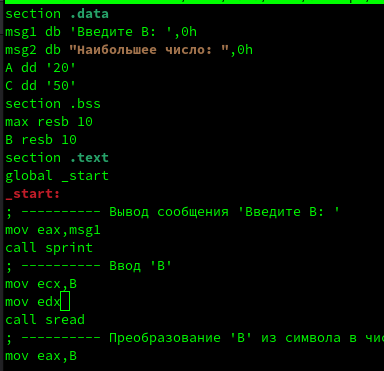
Разберём несколько строк файла листинга:

1)Строка под номером 14 перемещает содержимое msg1 в регистр eax. Адрес указывается сразу после номера. Следом идёт машинный код, который представляет собой исходную ассемблированную строку в виде шестнадцатиричной системы. Далее идёт исходный код

2)15-ая строка отвечает за вызов функции sprint. Она также имеет адрес и машинный код

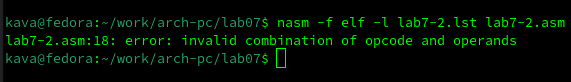
3)Строка 17 отвечает за запись переменной B в регистр ecx. Как видно, все строки имеют номер, адрес, машинный код и исходный код.

Теперь пробуем допустить ошибку убрав у команды move 1 операнд.



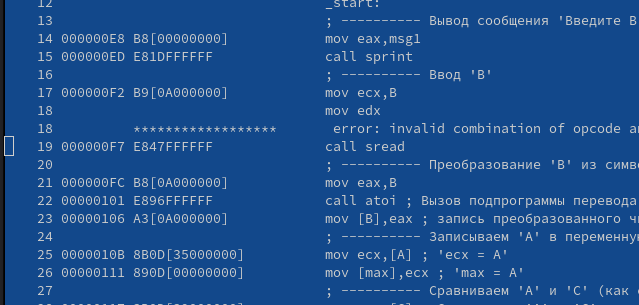
15

Пробуем собрать файл с ошибками.



16

Теперь зайдем в файл листинг и посмотрим отображается ли в нём ошибка.



17

Как видим появилась ошибка

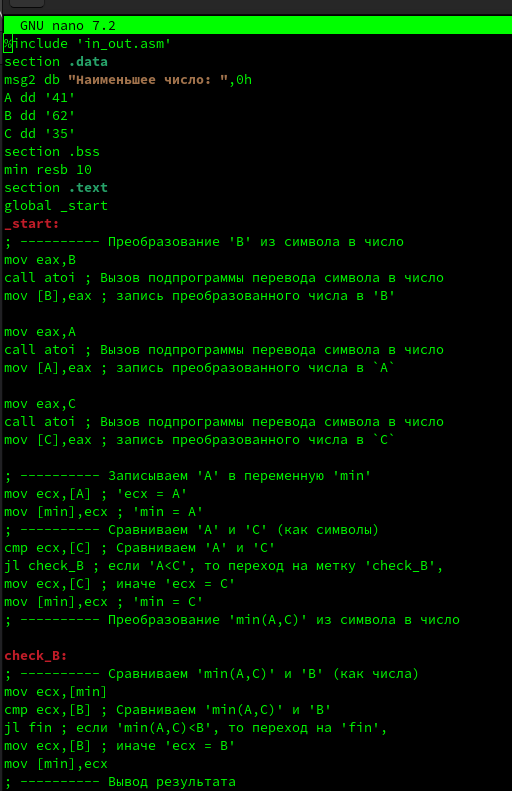
# 3 Выполнение заданий для самостоятельной работы

Создадим файл для выполнения самостоятельной работы. Мой вариант-10

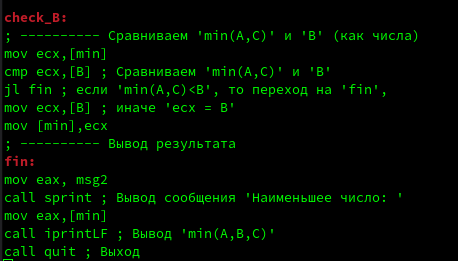
18

18

Напиши код для выполнения задания.

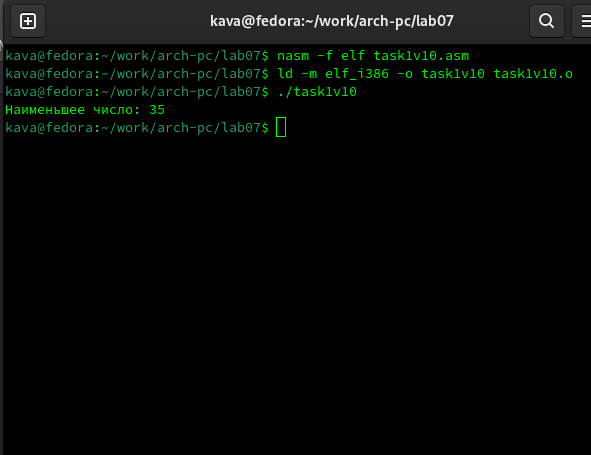


19



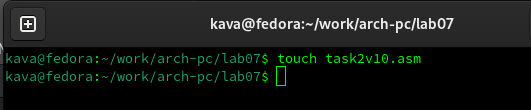
20

Соберем и запустим его.



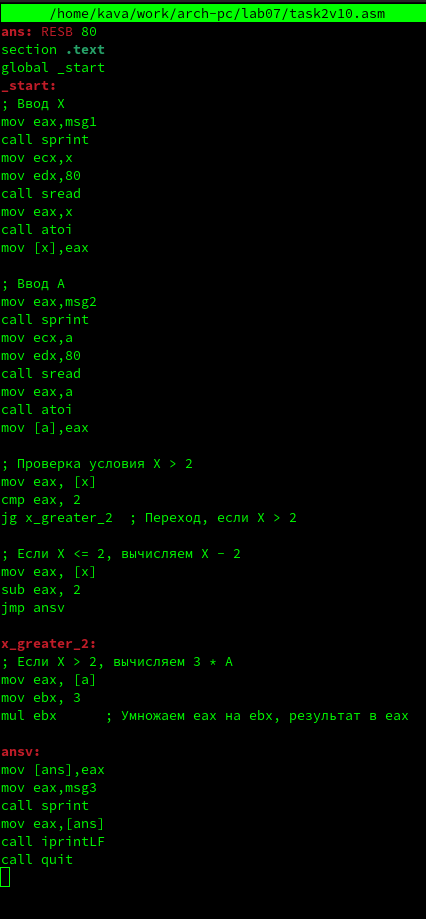
21

Теперь создаем второй файл для второго задания.



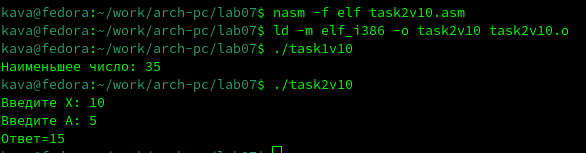
22

И напишем следующий код.



23

Соберём и запустим этот файл.



24

Как видим программа всё посчитала правильно.

# 4 Выводы

После выполнения лаборатоной работы. Я изучил команды условного и безусловного перехода. Приобрёл навыки написания программ с использованием переходов, познакомился с назначением и структурой файла листинга