Отчёт по лабораторной работе №7

дисциплина: Архитектура вычислительных систем

Мосолов Александр Денисович

Содержание

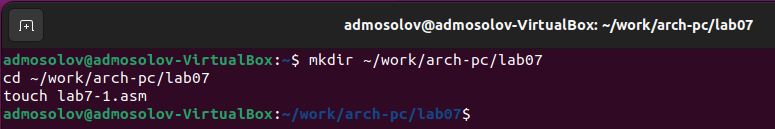
# 1 Цель работы

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

# 2 Выполнение лабораторной работы

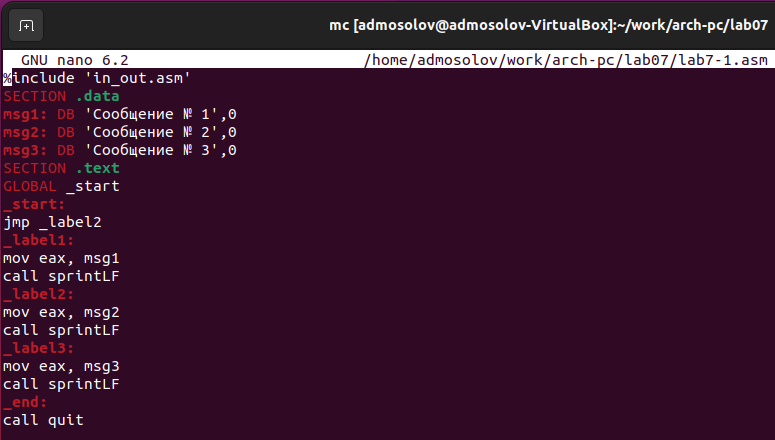
## 2.1 Реализация переходов в NASM

Создаём каталог для программ лабораторной работы № 7, переходим в него и создаём файл *lab7-1.asm*.



Создаем каталог для работы

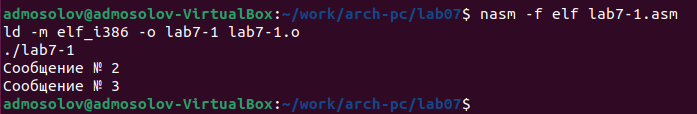
Инструкция *jmp* в *NASM* используется для реализации безусловных переходов. Рассмотрим пример программы с использованием инструкции *jmp*. Введите в файл *lab7-1.asm* текст программы из листинга *7.1*.



Текст программы в соттветствии с листингом 7.1

Для того, чтобы программа транслировалась без ошибок перенесем файл *in\_out.asm* в *~/work/arch-pc/lab06*.

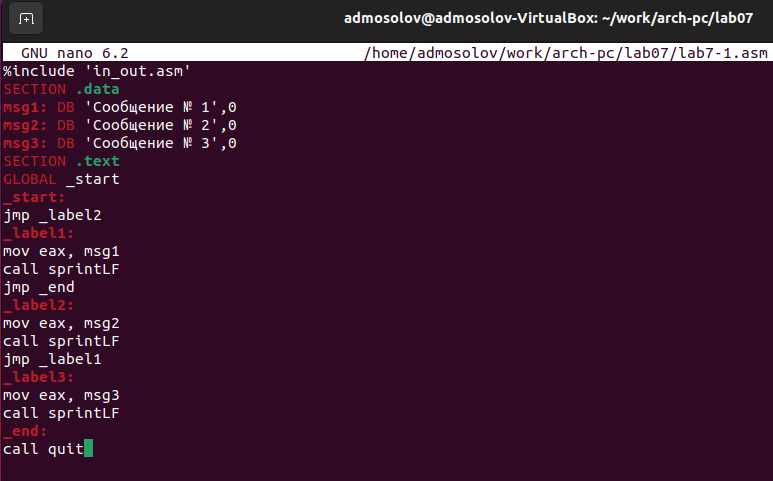
Создадим исполняемый файл и запустим его. Результат работы данной программы будет следующим:



Вывод программы из листинга 7.1

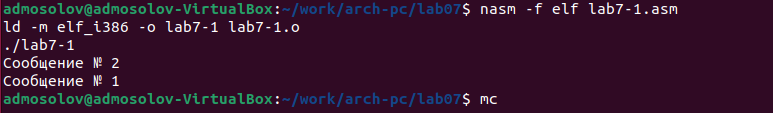
Таким образом, использование инструкции \*jmp \_label2\* меняет порядок исполнения инструкций и позволяет выполнить инструкции начиная с метки \*\_label2\*, пропустив вывод первого сообщения.

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*). Изменим текст программы в соответствии с листингом *7.2*



Текст программы в соответствии с листингом 7.2

Создадим исполняемый файл и проверим его работу.



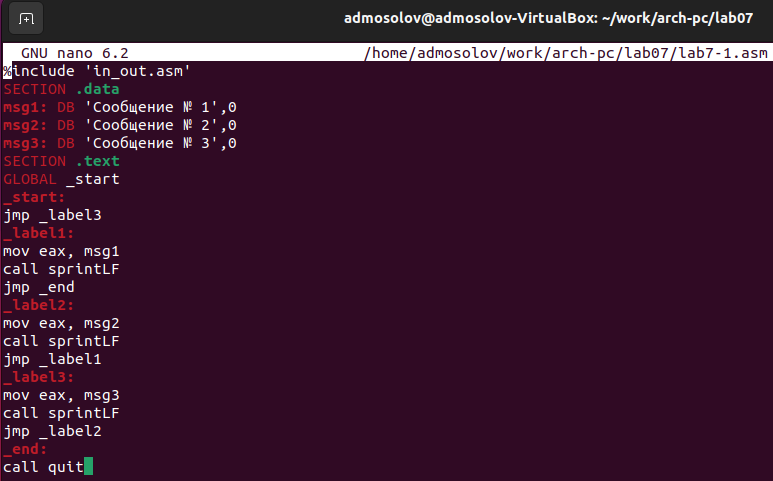
Вывод программы из листинга 7.2

Изменим текст программы добавив или изменив инструкции *jmp*, чтобы вывод программы был следующим:

*Сообщение № 3*

*Сообщение № 2*

*Сообщение № 1*



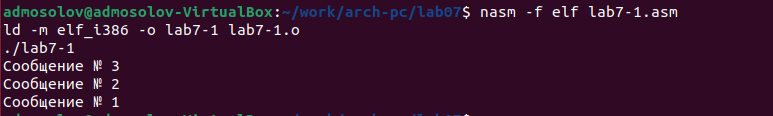
Текст программы lab7-1

Транслируем текст, создаём объектный файл, компилируем его и запускаем программу *lab7-1*:

*nasm -f elf lab7-1.asm*

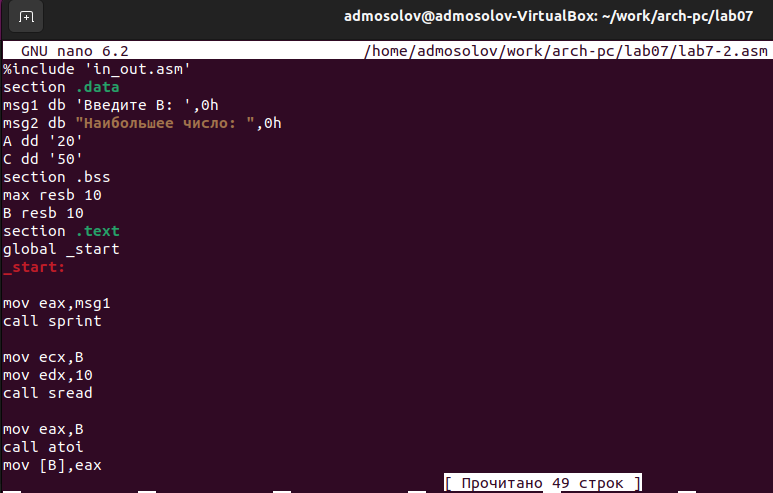
*ld -m elf\_i386 -o lab7-1 lab7-1.o*

*./lab7-1*



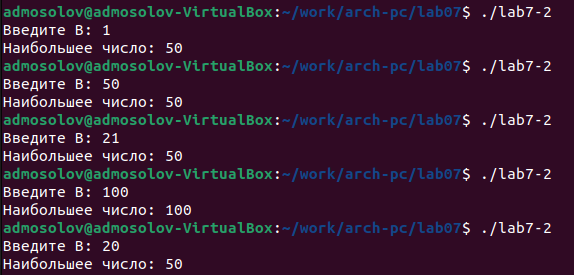
Вывод программы lab7-1

Создаём файл *lab7-2.asm* в каталоге *~/work/arch-pc/lab07*. Изучаем текст программы из листинга *7.3* и вводим его в *lab7-2.asm.*



Текст программы из листинга 7.3

Создаём исполняемый файл и проверяем его работу для разных значений *B*. Проверим работу программы для чисел: *1, 50, 21, 100, 20*.



Проверяем работу программы из листинга 7.3

## 2.2 Изучение структуры файлы листинга

Обычно nasm создаёт в результате ассемблирования только объектный файл. Получить файл листинга можно, указав ключ *-l* и задав имя файла листинга в командной строке. Создаём файл листинга для программы из файла *lab7-2.asm*. Вводим команду:

*nasm -f elf -l lab7-2.lst lab7-2.asm*

Файл листинга lab7-2.lst

Файл листинга lab7-2.lst

Откроем файл листинга *lab7-2.lst* с помощью текстового редактора *nano*.

Задание: внимательно ознакомиться с форматом листинга и содержимым. Подробно объяснить содержимое трёх строк файла листинга по выбору.

Строка 34: 0000001D - адрес в сегменте кода, BB01000000 - машинный код, mov ebx, 1 - присвоение ebx значение 1

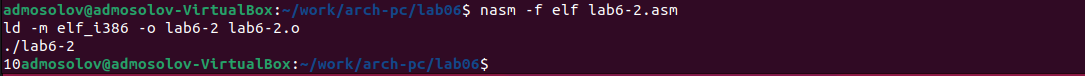
Строка 35: 00000022 - адрес в сегменте кода, B804000000 - машинный код, mov eax, 4 - присвоение eax значение 1

Строка 34: 00000027 - адрес в сегменте кода, CD80 - машинный код, int 80h - обращение к ядру



Текст файла листинга 7.3

А теперь на вывод программы с использованием *iprint*.



Используем iprint

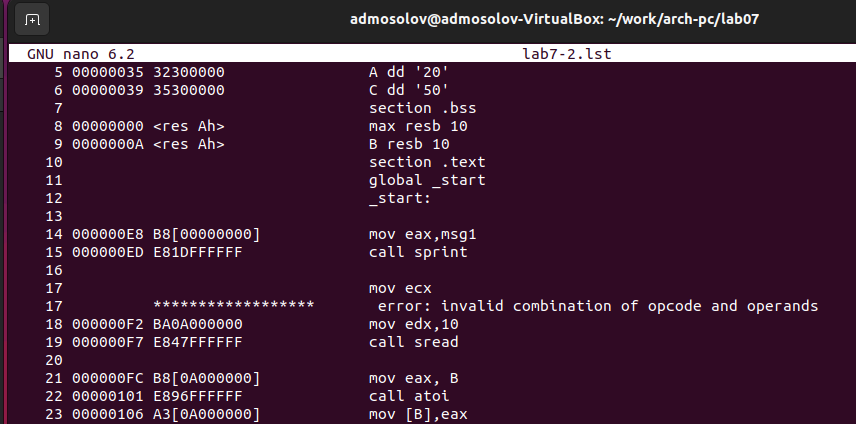
Задание: откройте файл с программой *lab7-2.asm* и в любой инструкции с двумя операндами удалить один операнд. Выполните трансляцию с получением файла листинга.

Проверим файл изначального листинга.

Листинг lab7-2.lst

Листинг lab7-2.lst

Уберём из него один операнд - *B* в 17 строке. Посмотрим файл листинга после повторного получения.



Текст листинга после удаления из текста программы операнда B

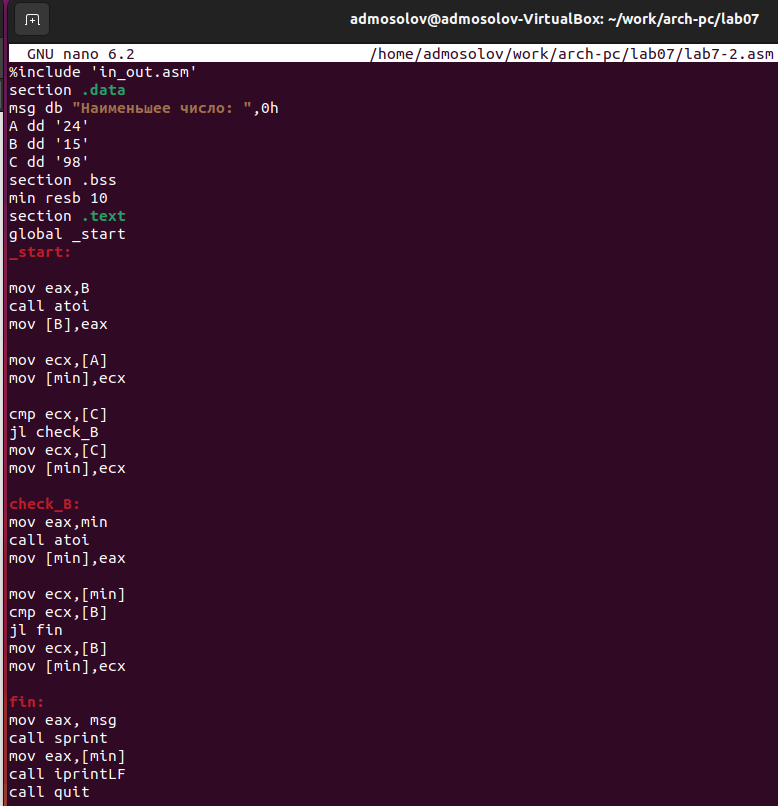
В листинге добавляется текст, сообщающий об ошибке. Выходные данные: текст об ошибке компиляции.

## 2.3 Задание для самостоятельной работы

Напишите программу нахождения наименьшей из *3* целочисленных переменных *a, b, 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.

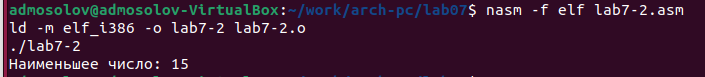
В моём случае вариант *9* - сравниние чисел *15, 24, 98*.

Запишем текст программы в *lab7-2.asm*



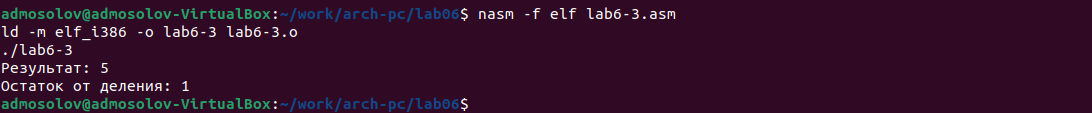
Текст программы нахождения наименьшего из трёх чисел

Проверим программу на работоспособность. Транслируем полученный текст программы *lab7-2.asm* в объектный файл. Выполняем компоновку объектного файла и запускаем получившийся исполняемый файл.



Результат работы программы поиска наименьшего из трёх чисел

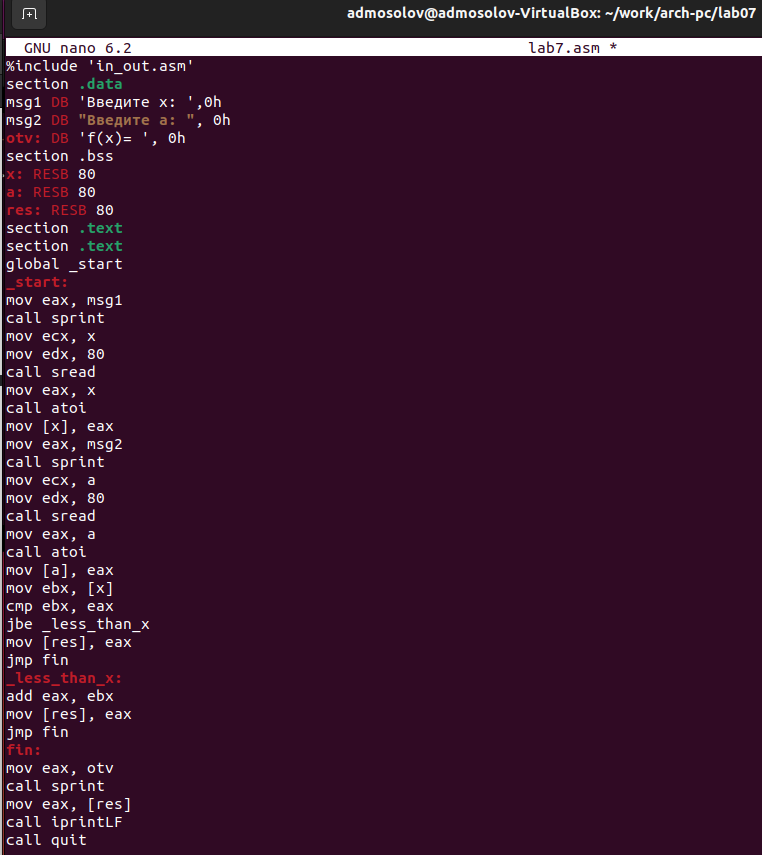
Транслируем полученный текст программы *lab6-3.asm* в объектный файл. Выполняем компоновку объектного файла и запускаем получившийся исполняемый файл.



Результат запуска изменённого файла lab6-3

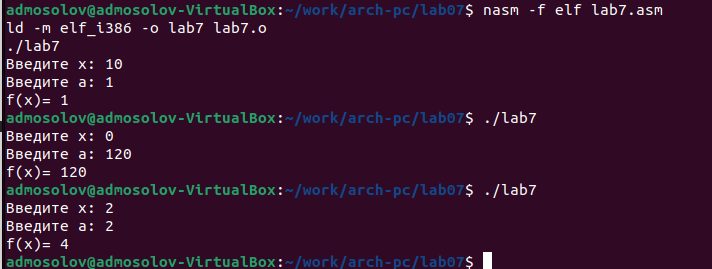
Напишите программу, которая для введенных с клавиатуры значений *x* и *a* вычисляетзначение заданной функции *f(x)* и выводит результат вычислений. Вид функции *f(x)* выбрать из таблицы *7.6* вариантов заданий в соответствии с вариантом, полученным при выполнении лабораторной работы *№ 6*. Создайте исполняемый файл и проверьте его работу для значений *x* и *a* из *7.6*.

Создадим файл *lab7.asm*, напишем текст программы.



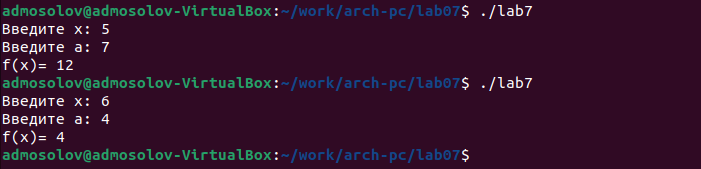
Текст программы для вычисления *f(x)*

Транслируем текст программы *lab7.asm* в объектный файл. Выполняем компоновку объектного файла и запускаем получившийся исполняемый файл.



Результат работы программы по вычислению *f(x)*

Проверим работу программы подставив *a=5*, *x=7* и *a=6*, *x=4*.



Проверка программы для a и x из таблицы

# 3 Выводы

В ходе лабораторной работы были изучены команды условного и безусловного переходов. Приобретены навыки написания программ с использованием переходов. Помимо этого, была усвоена информация, связанная с назначением и структурой файла листинга.