РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7

дисциплина: Архитектура компьютера

Студент: Сухова Арина

Группа: НПИбд-02-25

МОСКВА

2025 г.

1. Цель работы
   Освоить условный и безусловный переход. Ознакомиться с назначением и структурой файла листинга.

2. Задание
   Написать программы для решения системы выражений.

3. Выполнение лабораторной работы
   3.1 Реализация переходов в NASM
   Создаем каталог для программ ЛБ7 и в нем создаем файл (рис.3.1)

```
aesukhova@fedora:~$ mkdir ~/work/arch-pc/lab07
aesukhova@fedora:~$ cd ~/work/arch-pc/lab07
aesukhova@fedora:~/work/arch-pc/lab07$ touch lab7-1.asm
```

   Рис. 3.1 Создаем каталог с помощью команды mkdir и файл с помощью команды touch

   Открываем файл в Midnight Commander и заполняем его в соответствии с листингом 7.1 (рис. 3.2)

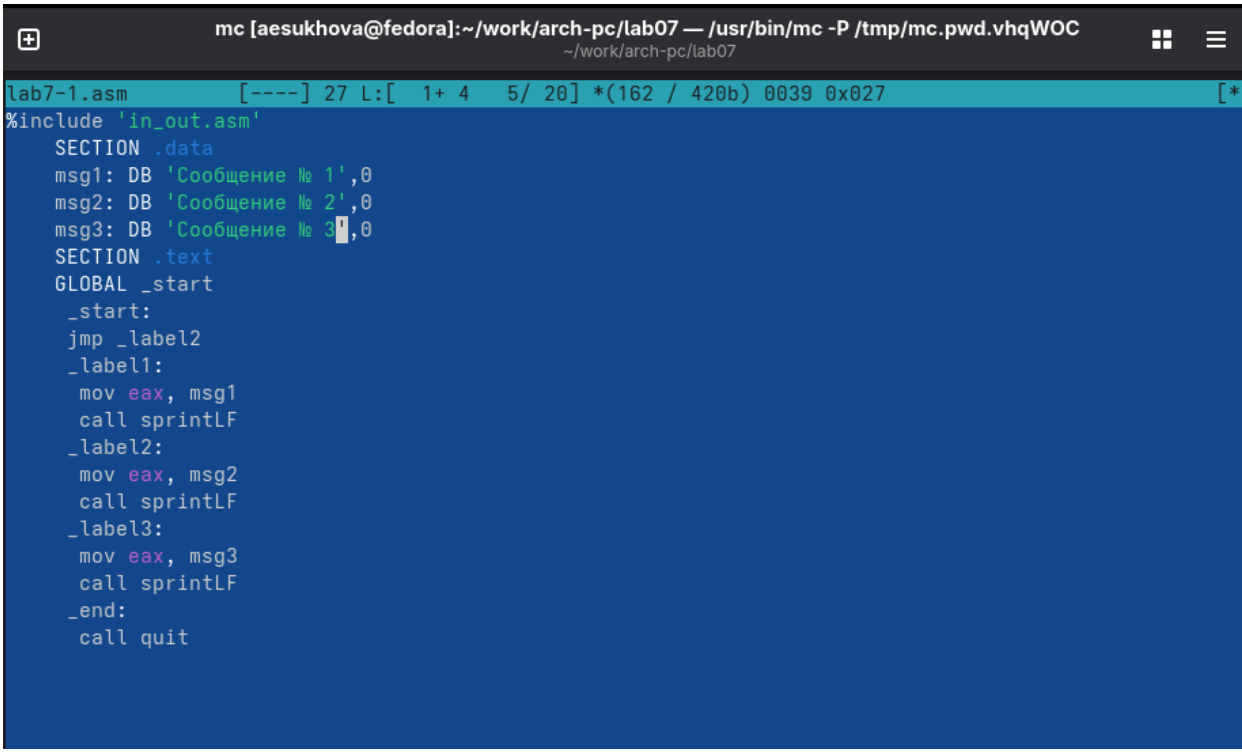

   Рис. 3.2 Заполняем файл

   Создаем исполняемый файл и запускаем его (рис. 3.3)

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
aesukhova@fedora:~/work/arch-pc/lab07$
```

Рис. 3.3 Запускаем файл и смотрим на его работу

Снова открываем файл для редактирования и изменяем его в соответствии с листингом 7.2 (рис. 3.4)

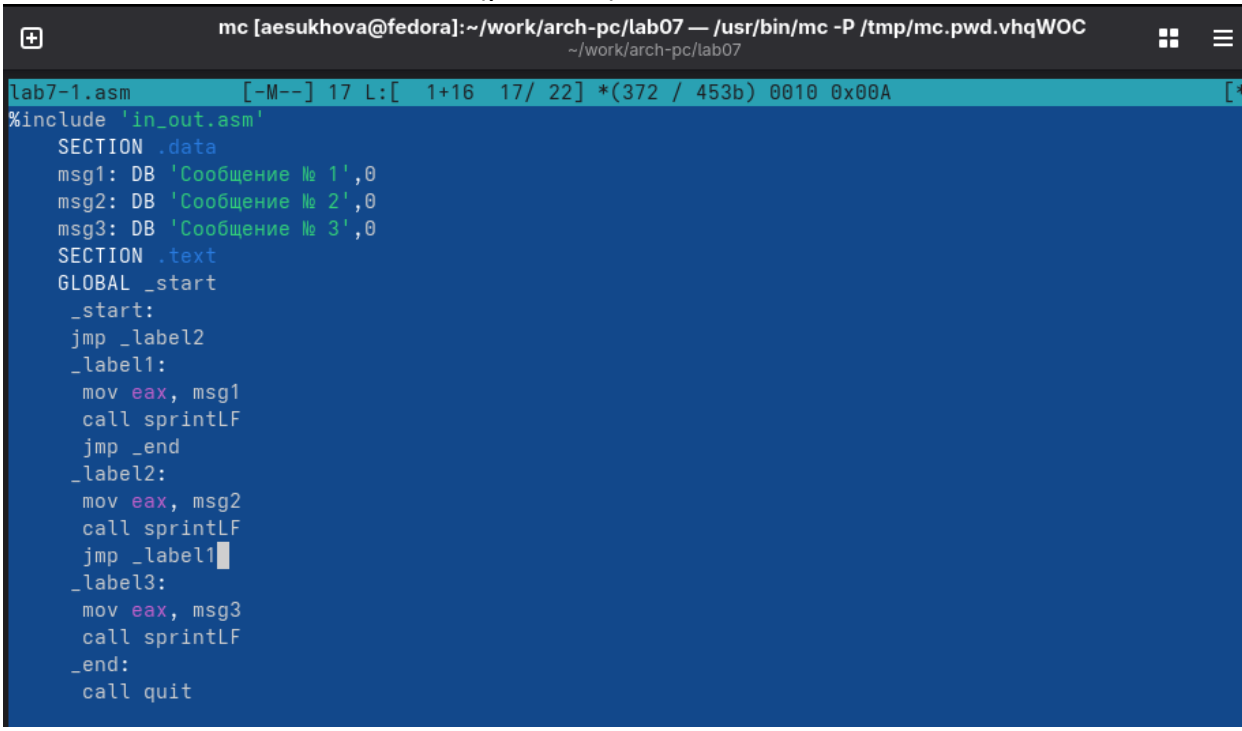

Рис. 3.4 Изменяем файл

Создаем исполняемый файл и запускаем его (рис. 3.5).

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 1
aesukhova@fedora:~/work/arch-pc/lab07$
```

Рис. 3.5 Запускаем файл и смотрим на его работу

Снова открываем файл для редактирования и изменяем его чтобы произошел данный вывод (рис. 3.6)

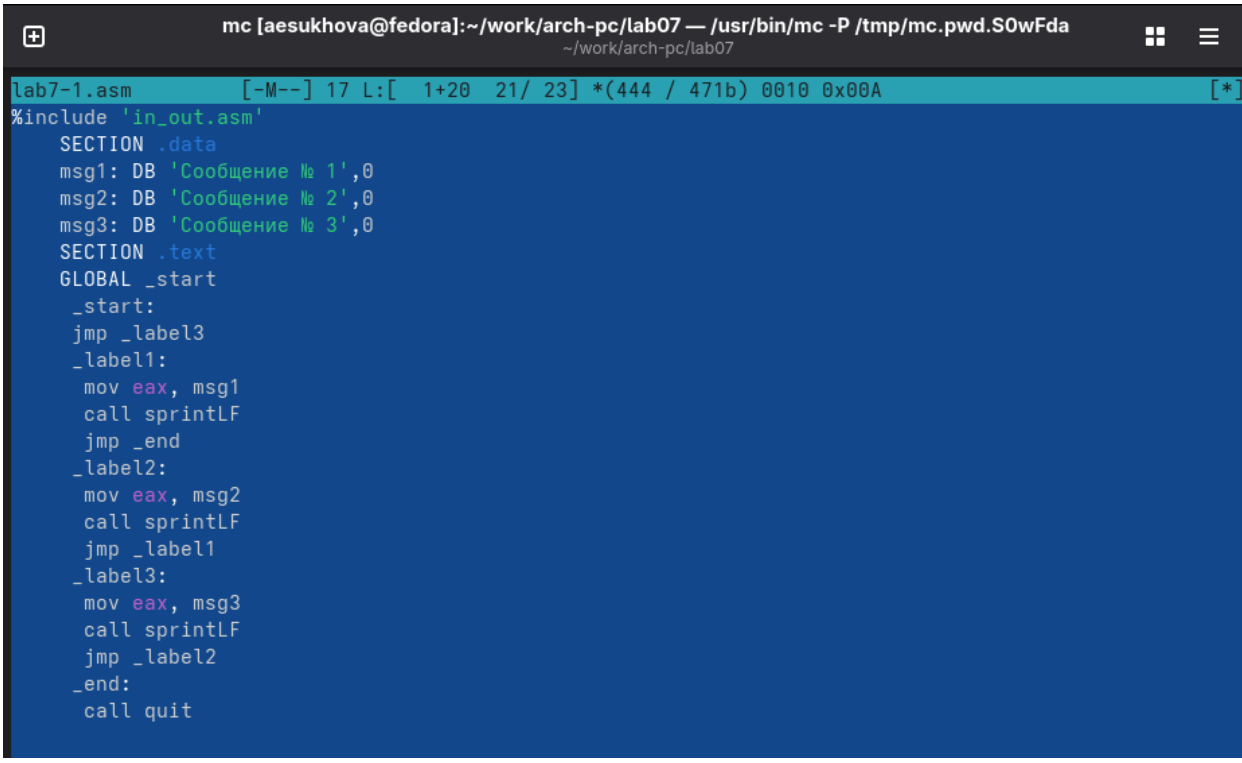

Рис. 3.6 Редактируем файл

Создаем исполняемый файл и запускаем его (рис. 3.7).

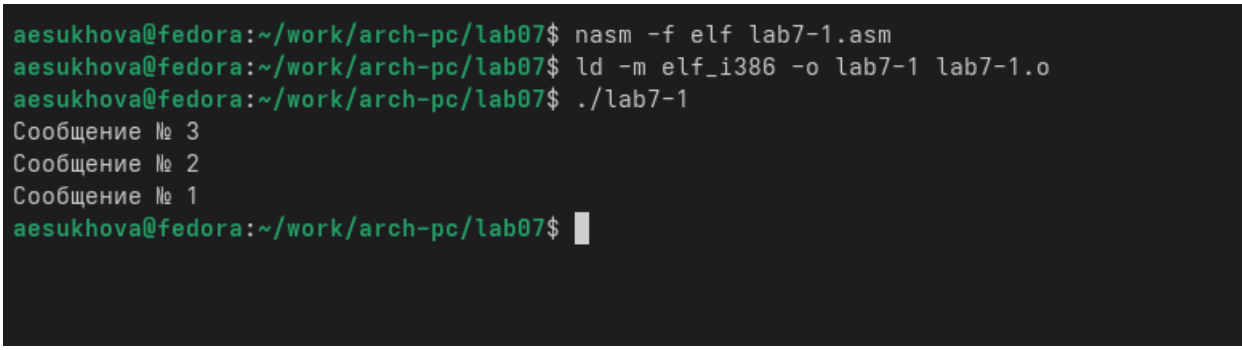

Рис. 3.7 Проверяем, сошелся ли наш вывод с данным в условии выводом

Создаем новый файл (рис. 3.8)

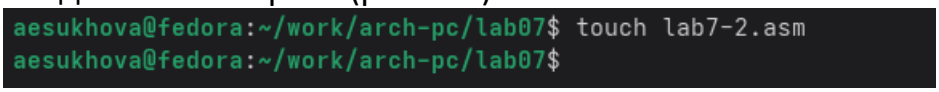

Рис. 3.8 Создаем файл командой touch

Открываем файл в Midnight Commander и заполняем его в соответствии с листингом 7.3 (рис. 3.9)

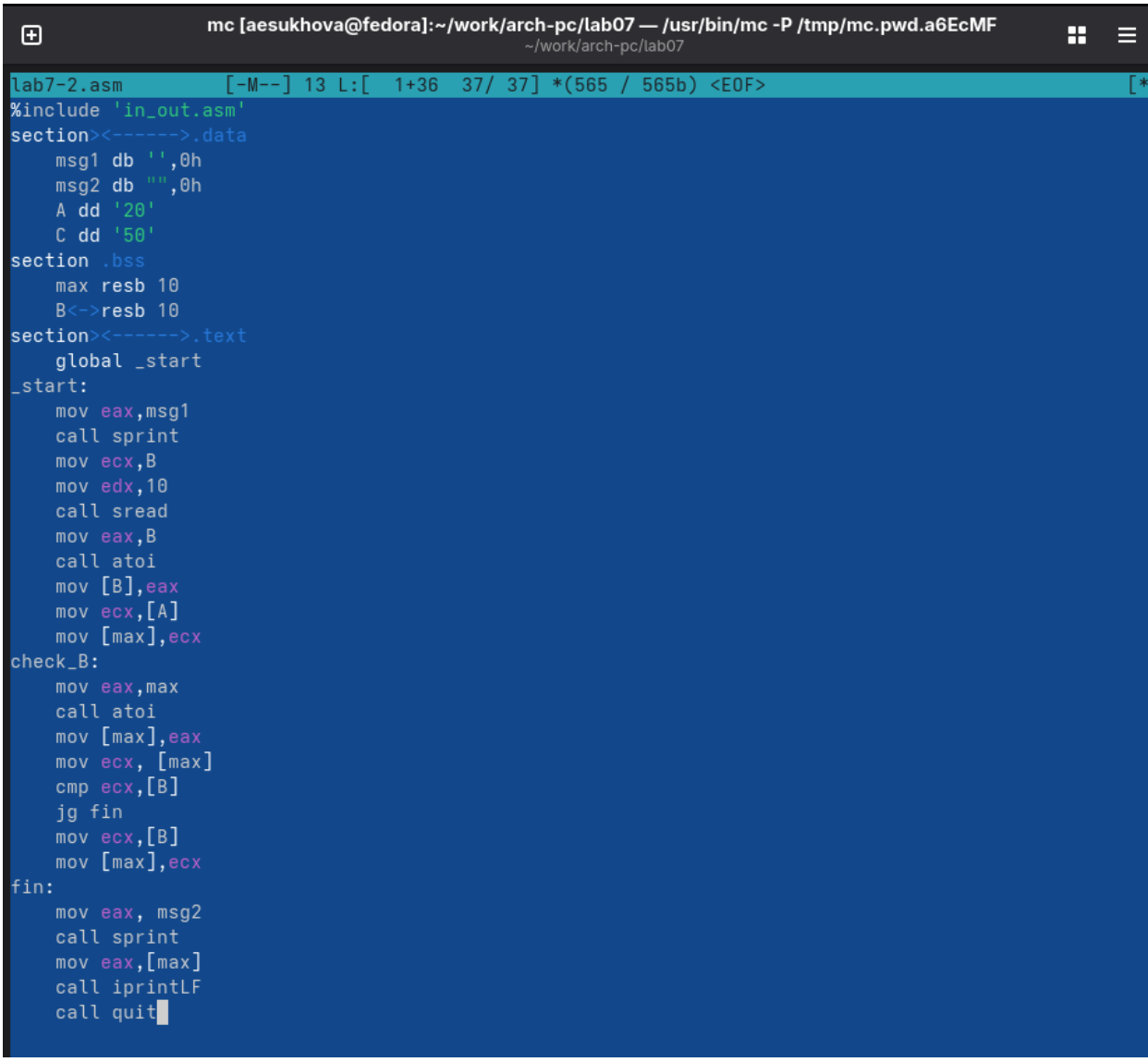

Рис. 3.9 Заполняем файл

Создаем исполняемый файл и проверяем его работу, вводя разные значения B (рис. 3.10).

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-2
5
20
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-2
10
20
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-2
1
20
aesukhova@fedora:~/work/arch-pc/lab07$
```

Рис. 3.10 Смотрим на работу программ

3.2 Изучение структуры файла листинга

Создаем файл листинга для программы lab7-2.asm (рис. 3.11)

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
aesukhova@fedora:~/work/arch-pc/lab07$
```

Рис.3.11 Создаем файл листинга

Открываем файл листинга с помощью команды mcedit и изучаем его (рис. 3.12)

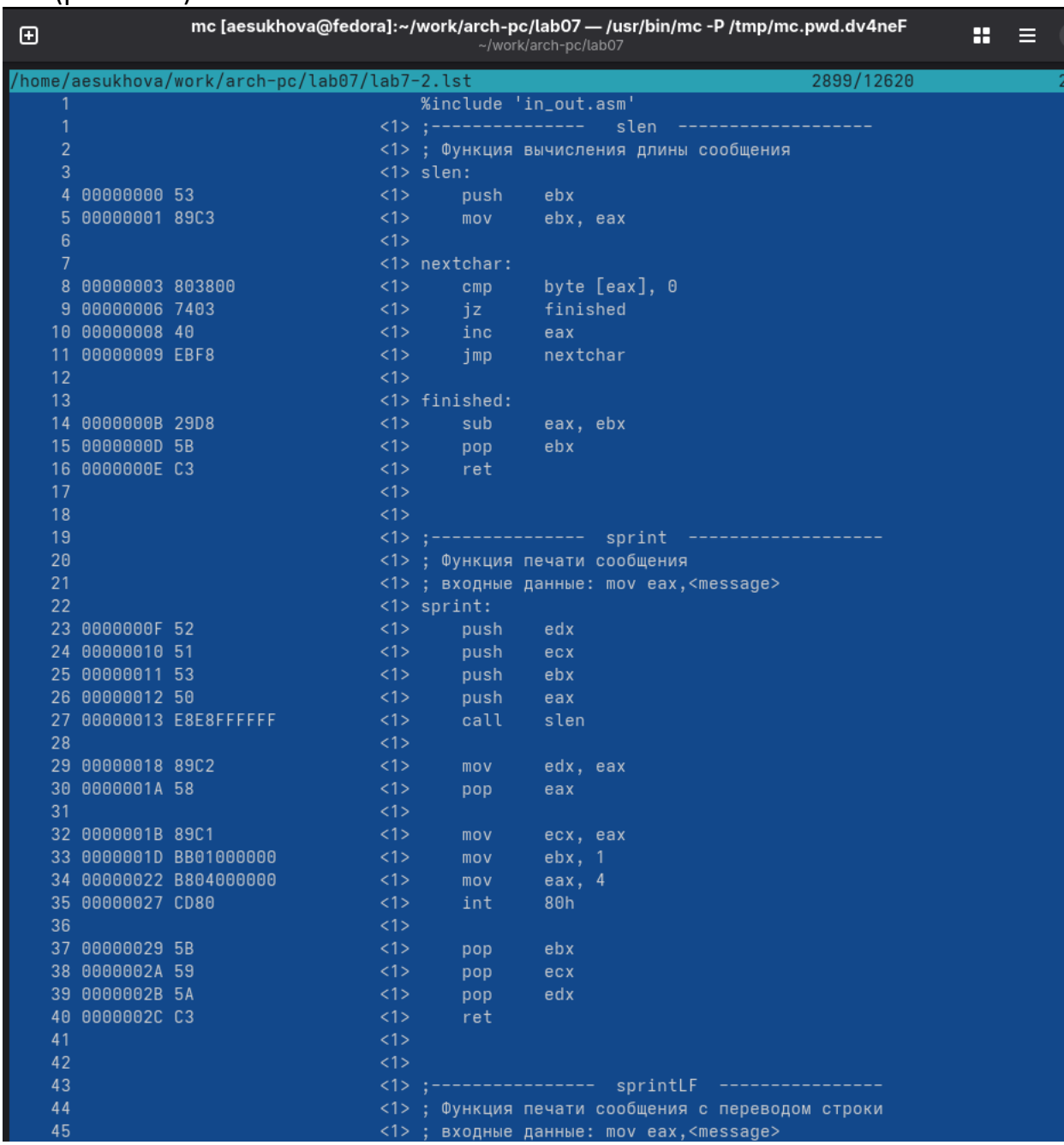

Рис. 3.12 Изучаем файл

Строка 33: 0000001D-адрес в сегменте кода,BB01000000-машинный код,mov ebx,1-присвоение переменной ecx значения 1.
Строка 34: 00000022-адрес в сегменте кода,B804000000-машинный код,mov eax,4-присвоение переменной eax значения 4.

Строка 35 00000027-адрес в сегменте кода,CD80-машинный код,int 80h-вызов ядра.

Открываем файл и удаляем один операндум (рис.3.13).

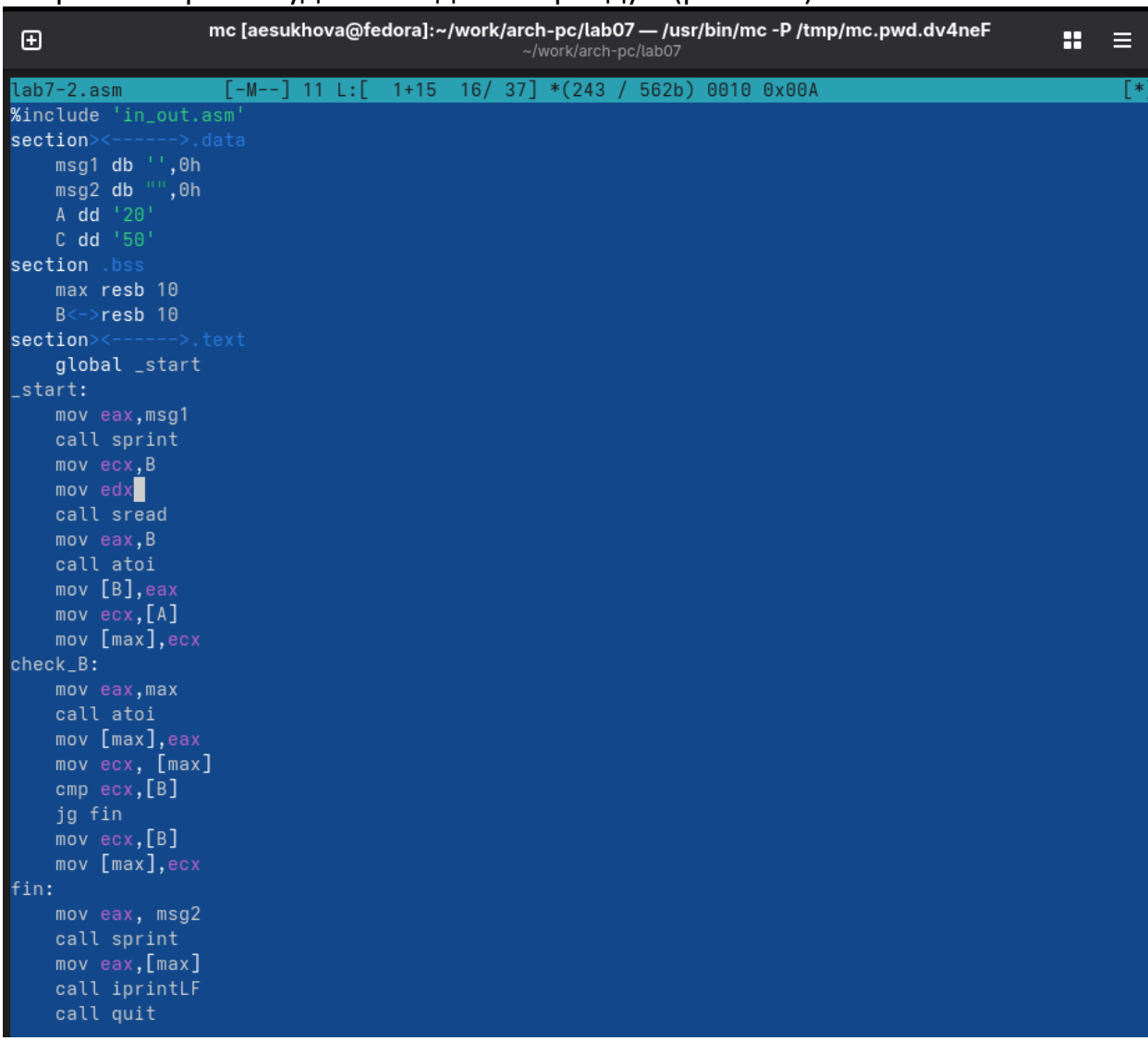

Рис. 3.13 Удаляем операндум из файла

Транслируем с получением файла листинга (рис. 3.14).

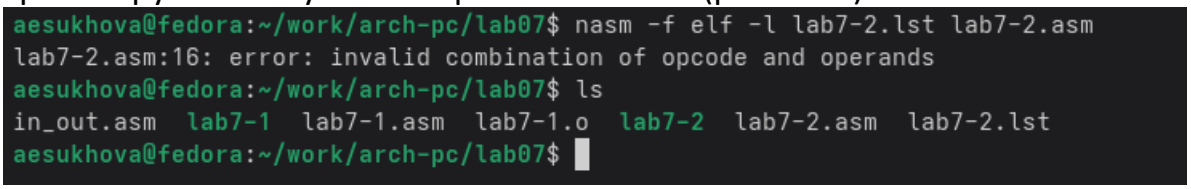

Рис. 3.14 Транслируем файл

При трансляции файла выдается ошибка, но создаются исполнительный файл lab7-2 и lab7-2.lst
Снова открываем файл листинга и изучаем его (рис. 3.15).

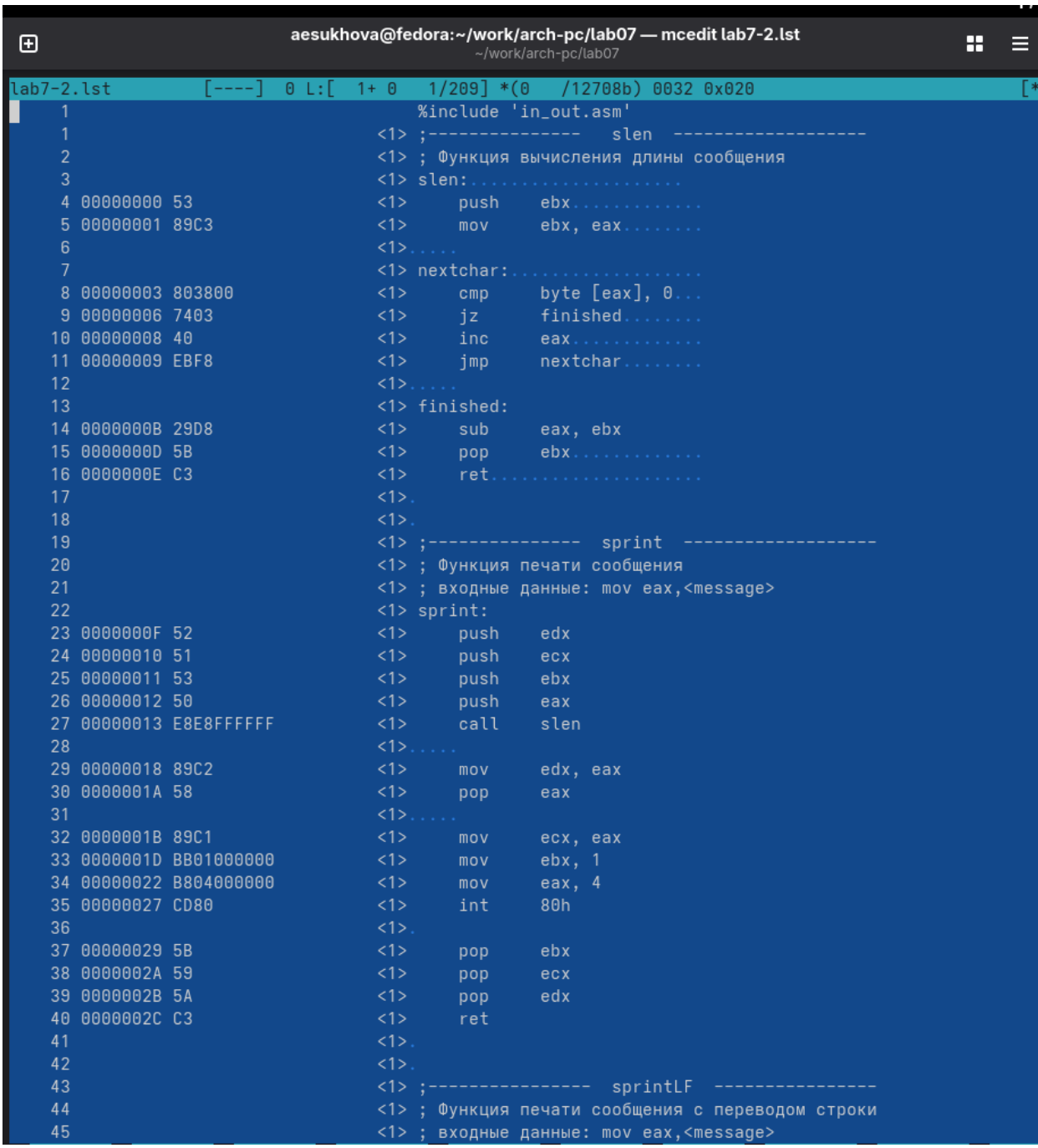

Рис. 3.15 Анализируем файл с ошибкой

Создается ТОЛЬКО файл листинга `lab7-2.lst`. Файл объектного кода lab7-2.o НЕ создается, потому что ассемблер обнаружил ошибку и остановил компиляцию.
При наличии синтаксических ошибок в исходном коде:
- Объектный файл (.o) не создается - компиляция прерывается
- Листинг создается, но содержит сообщения об ошибках - это помогает локализовать проблему
- Ассемблер указывает точное место и характер ошибки - в данном случае "ожидалась инструкция" из-за неполной команды

3.3 Задание для самостоятельной работы

ВАРИАНТ-5

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.

Создаем новый файл (рис. 3.16)

```
aesukhova@fedora:~/work/arch-pc/lab07$ touch lab7-3.1.asm
aesukhova@fedora:~/work/arch-pc/lab07$ mcedit lab7-3.1.asm
```

Рис. 3.16 Создаем файл командой touch и редактируем в Midnight Commander

Открываем его и пишем программу, которая выберет наименьшее число из трех (2 числа уже в программе, третье вводится с консоли)(рис. 3.17)

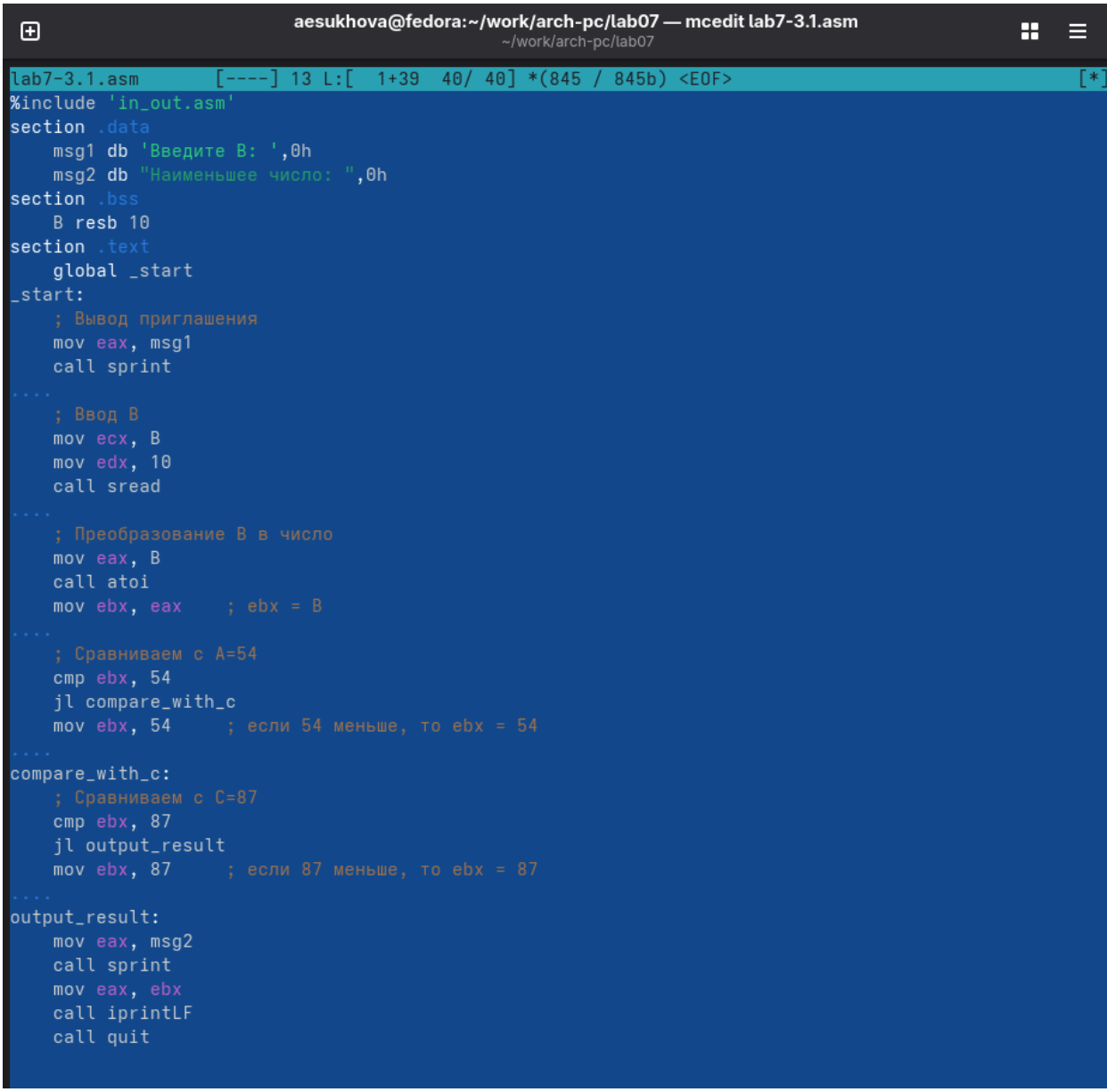

Рис. 3.17 Прописываем программу

Транслируем файл и смотрим на работу программы (рис. 3.18)

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-3.1.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-3.1 lab7-3.1.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-3.1
Введите B: 62
Наименьшее число: 54
```

Рис. 3.18 Смотрим на работу выполненного файла (всё верно)

2. 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 № 6. Создайте исполняемый файл и проверьте его работу для значений x и a из 7.6.

Создаем новый файл (рис. 3.19).

```
aesukhova@fedora:~/work/arch-pc/lab07$ touch lab7-4.asm
```

Рис. 3.19 Создаем файл командой touch

Открываем его и пишем программу, которая решит систему уравнений при данных, введенных в консоль (рис. 3.20)

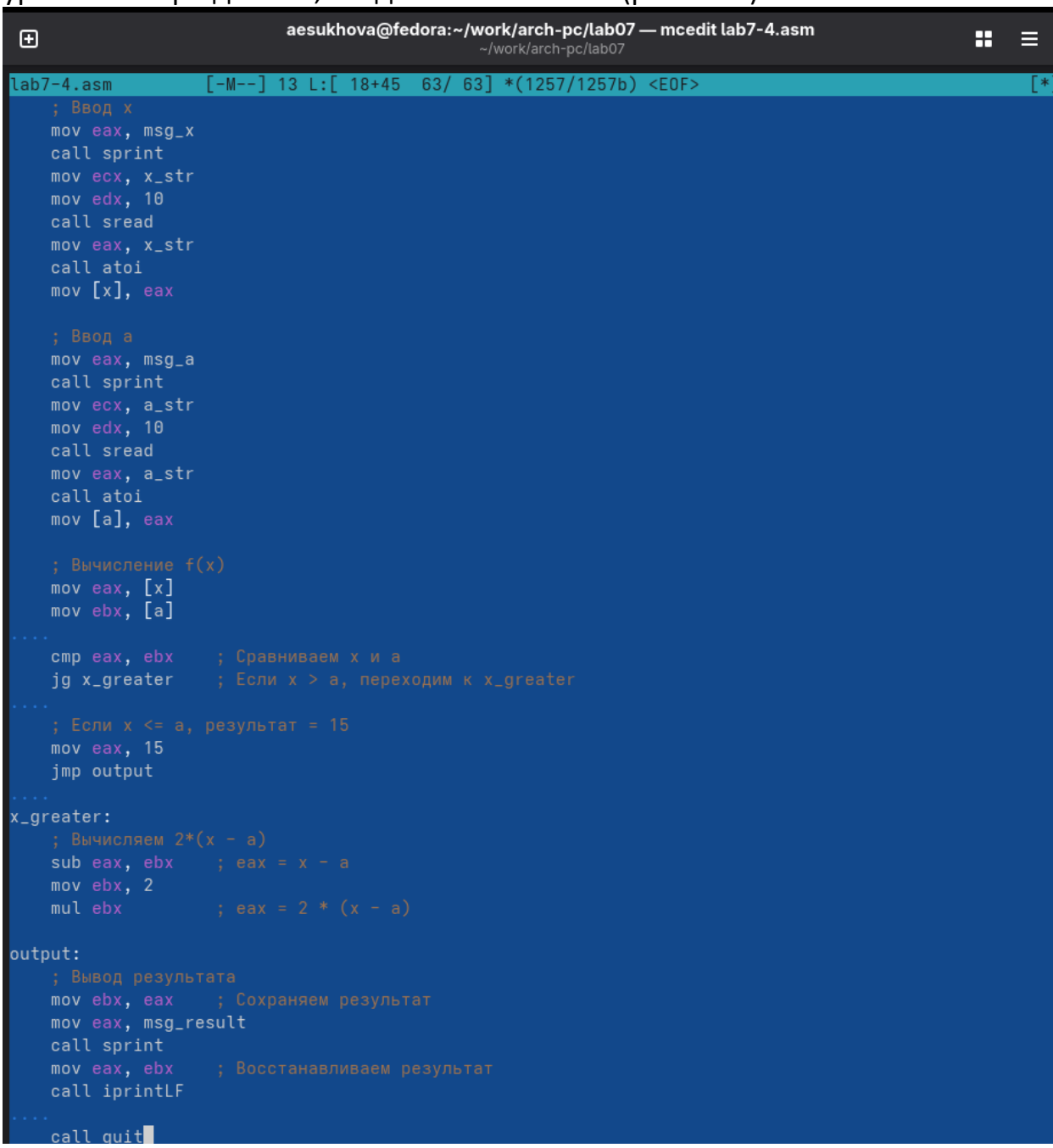

Рис. 3.20 Прописываем программу

Транслируем файл и проверяем его работу при x = 1 и a = 2 (рис. 3.21).

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите x: 1
Введите a: 2
Результат f(x): 15
```

Рис 3.21 Проверяем работу программы (работает верно)

Транслируем файл и проверяем его работу при x = 2 и a = 1 (рис 3.22)

```
aesukhova@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
aesukhova@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
aesukhova@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите x: 2
Введите a: 1
Результат f(x): 2
```

Рис. 3.22 Проверяем работу программы (работает верно)

4. Выводы

   Мы научились решать программы с использованием циклов и обработкой аргументов командной строки.