ОТЧЕТ по лабораторной работе №6

Полякова Юлия Александровна

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

# 2 Результаты выполнения лабораторной работы

1. Создаем каталог для программ лабораторной №6, переходим в него и создаем файл lab6-1.asm (Рис. 1).

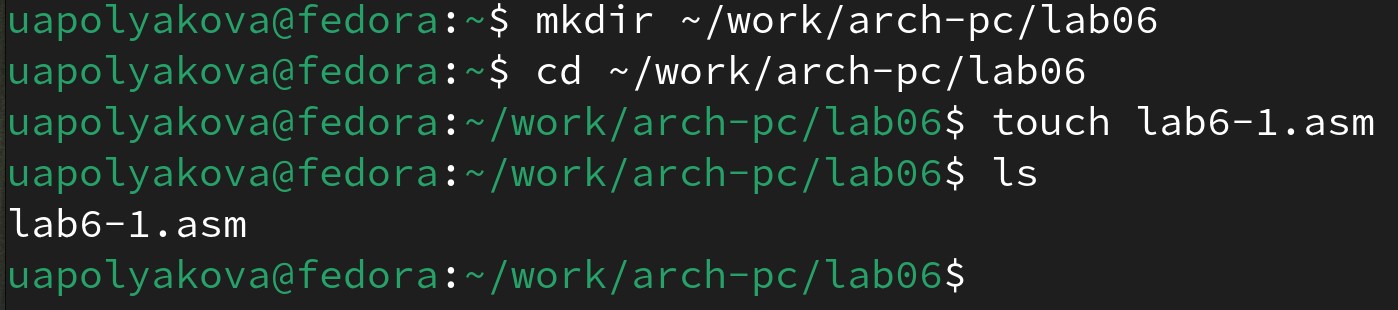


Рис. 1: Создание каталога и lab6-1.asm

1. Записываем в файл программу из листинга 6.1. (Рис. 2).

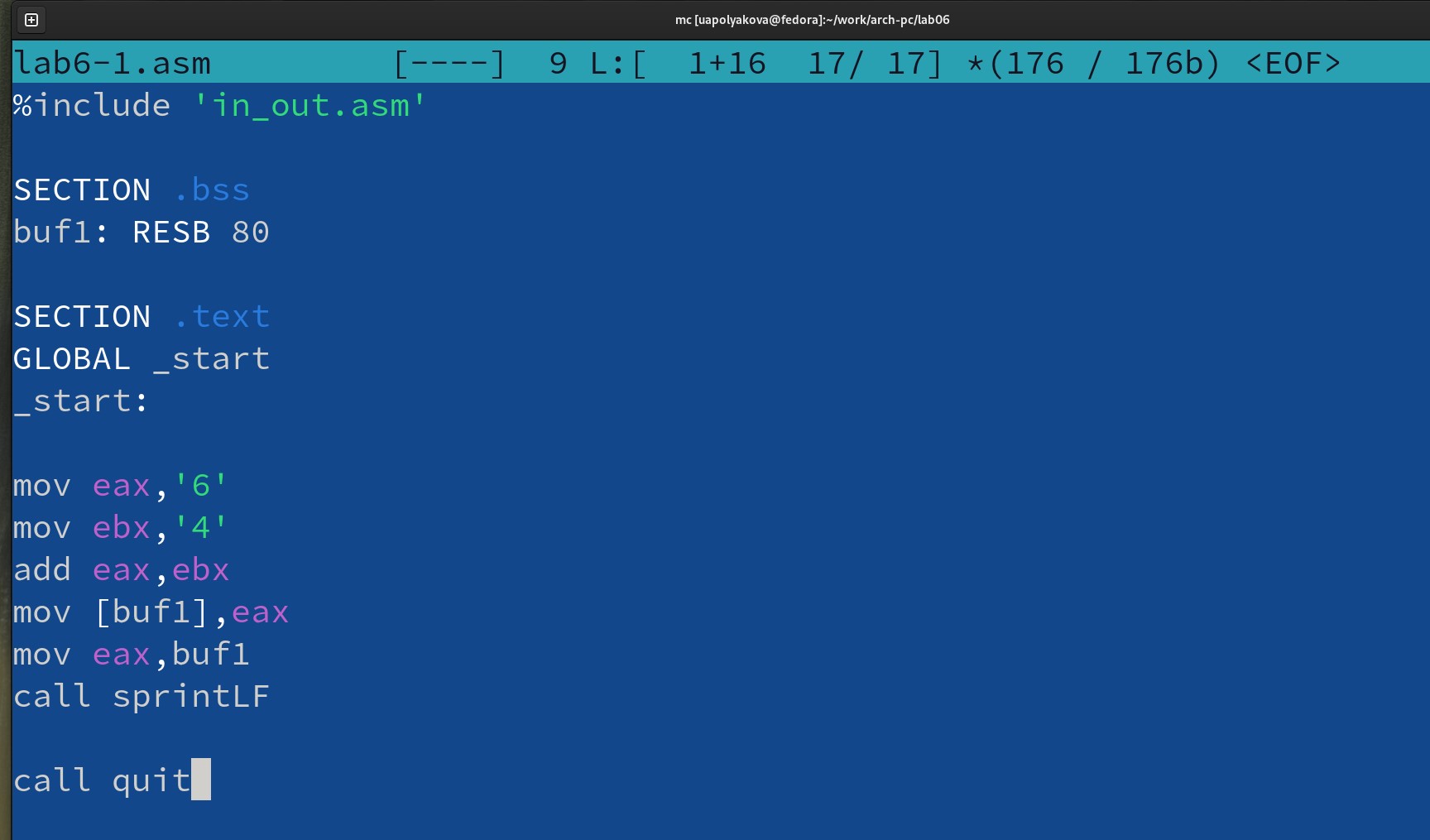


Рис. 2: lab6-1.asm из листинга 6.1.

1. Создаем исполняемый файл и запускаем его (Рис. 3).

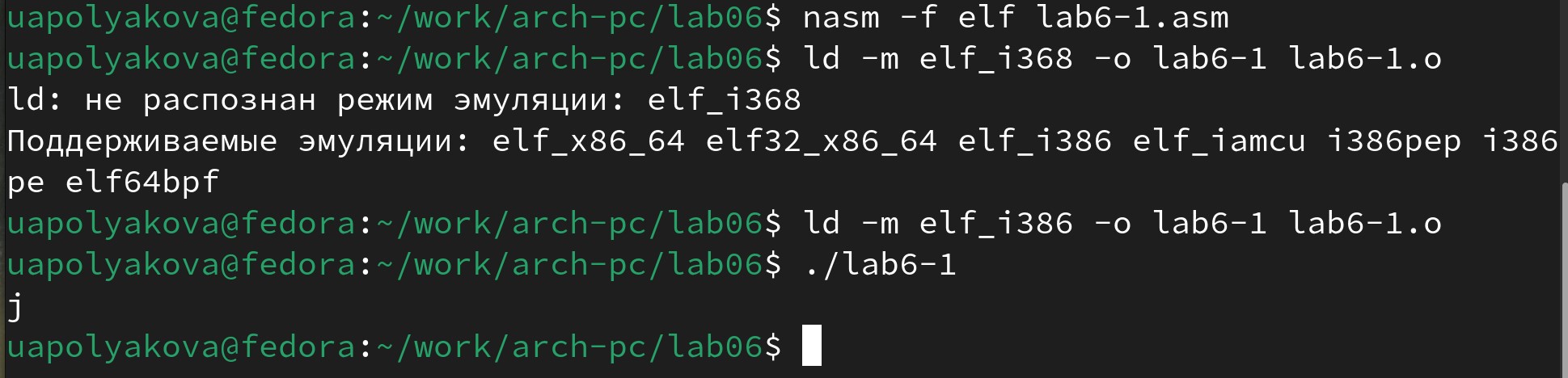


Рис. 3: Запуск lab6-1.asm из листинга 6.1.

1. Изменяем текст программы, вместо символов записываем в регистры числа (Рис. 4)

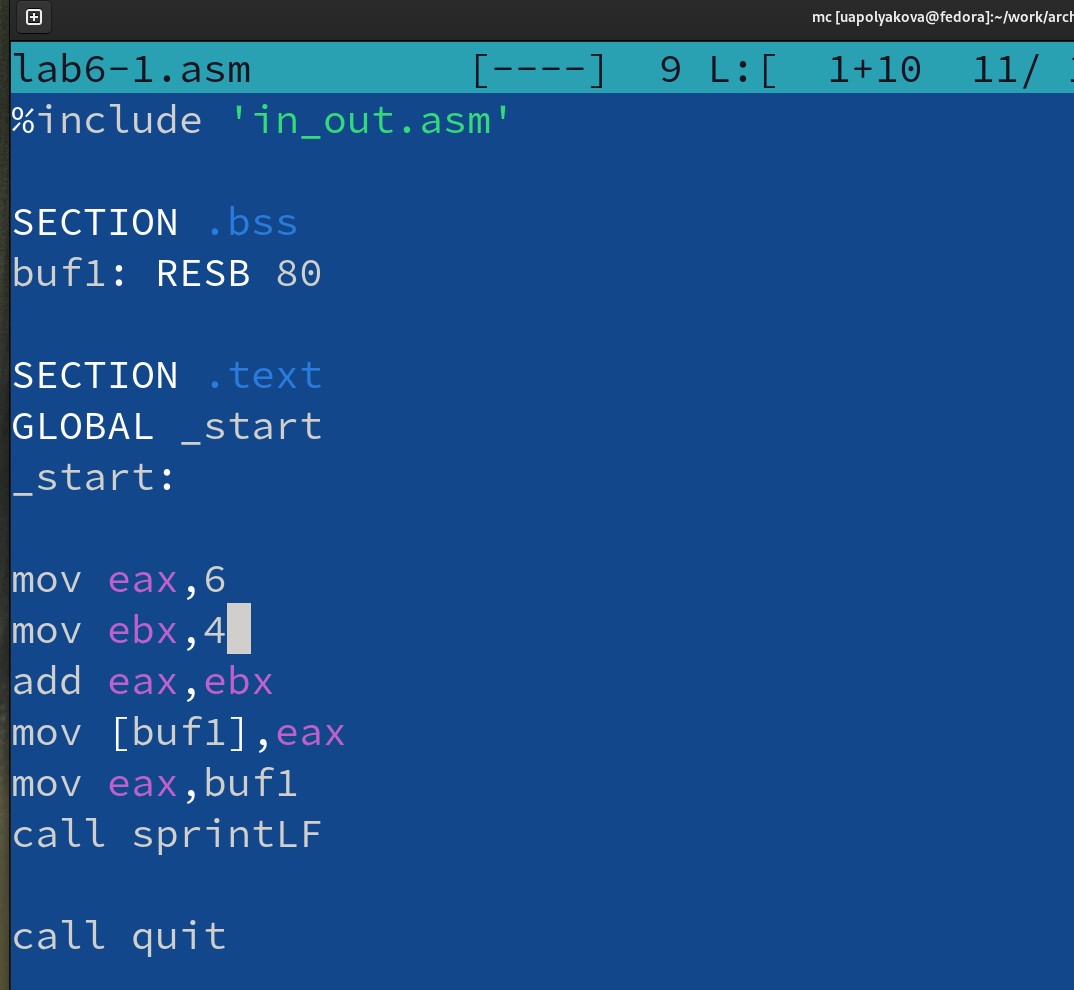


Рис. 4: Измененный lab6-1.asm

1. Создаем исполняемый файл и запускаем его (Рис. 5).

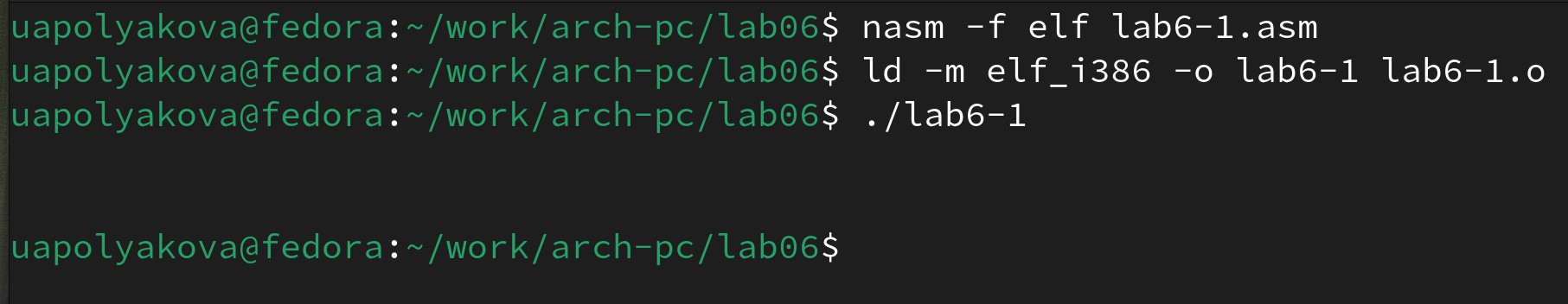


Рис. 5: Запукс измененного lab6-1.asm

По таблице ASCII это символ переноса строки \n (Рис. 6).

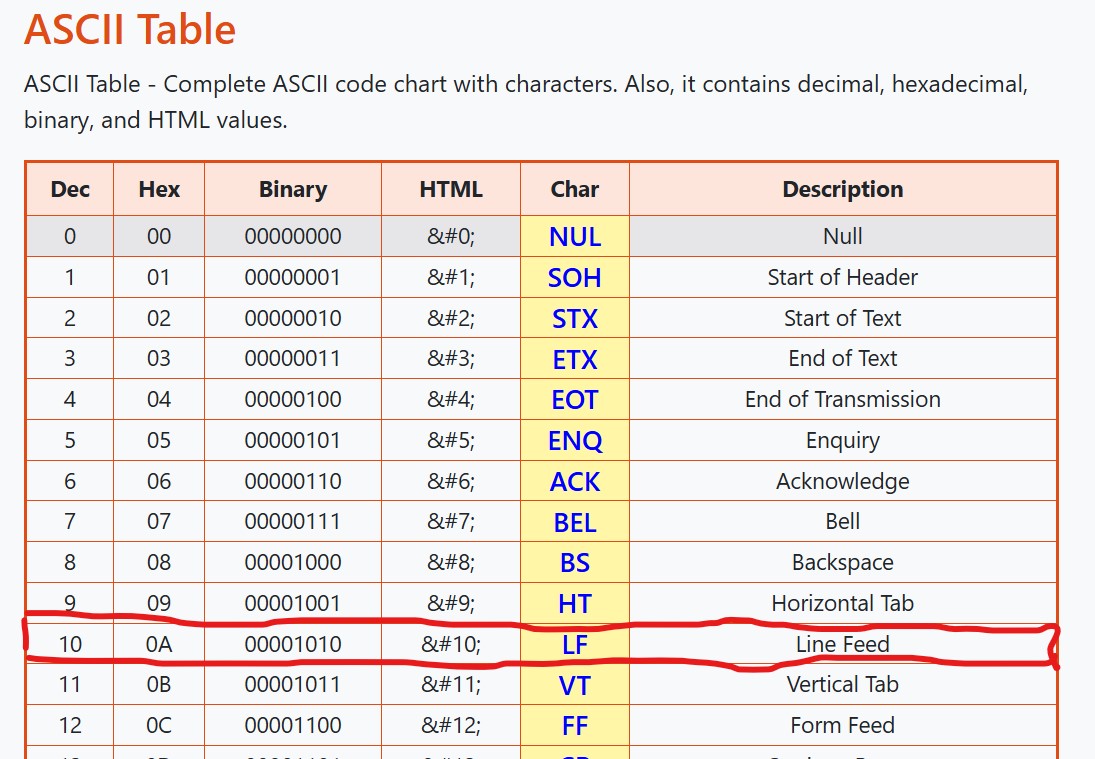


Рис. 6: Символ переноса строки

Он отображается при выводе на экран, так как видно, что после запуска программы образуются две пустые строки.

1. Создаем lab6-2.asm по листингу 6.2. (Рис. 7).

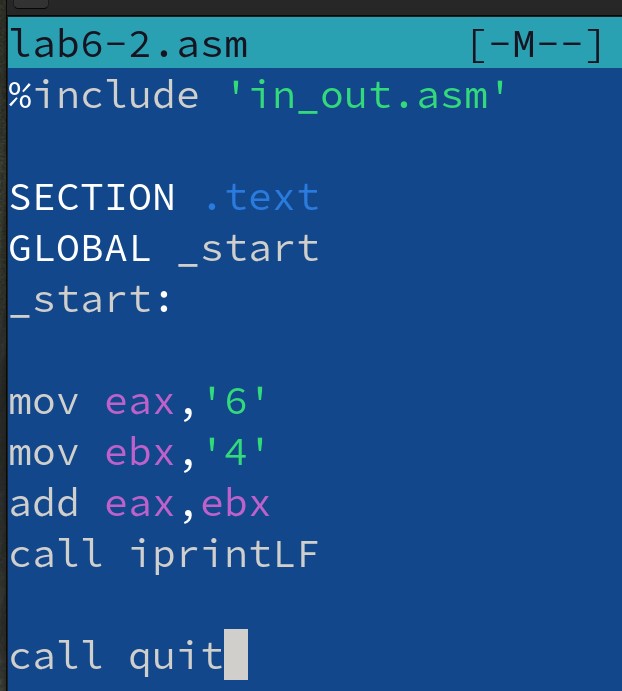


Рис. 7: lab6-2.asm из листинга 6.2.

1. Создаем исполняемый файл и запускаем его (Рис. 8).

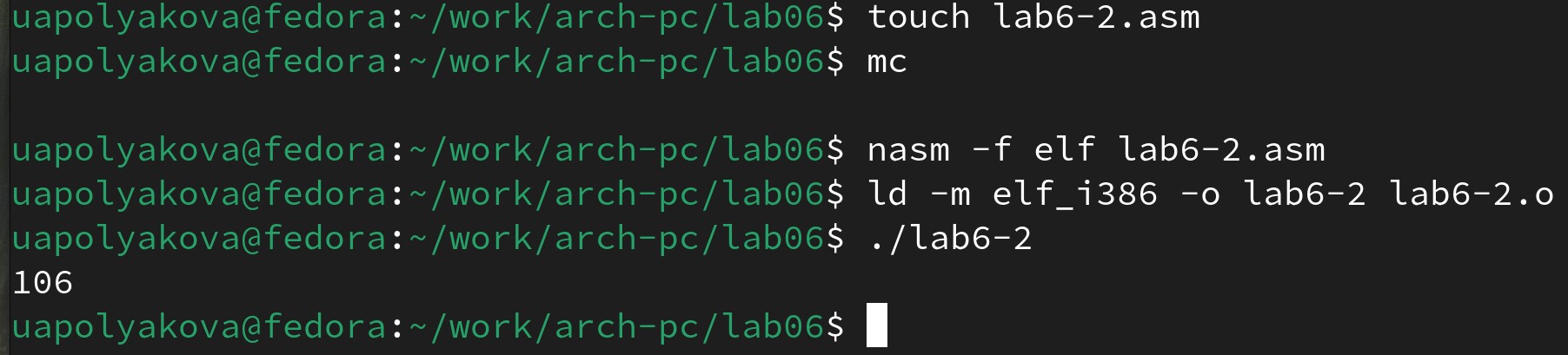


Рис. 8: Запуск lab6-2.asm из листинга 6.2.

1. Изменяем символы на числа.

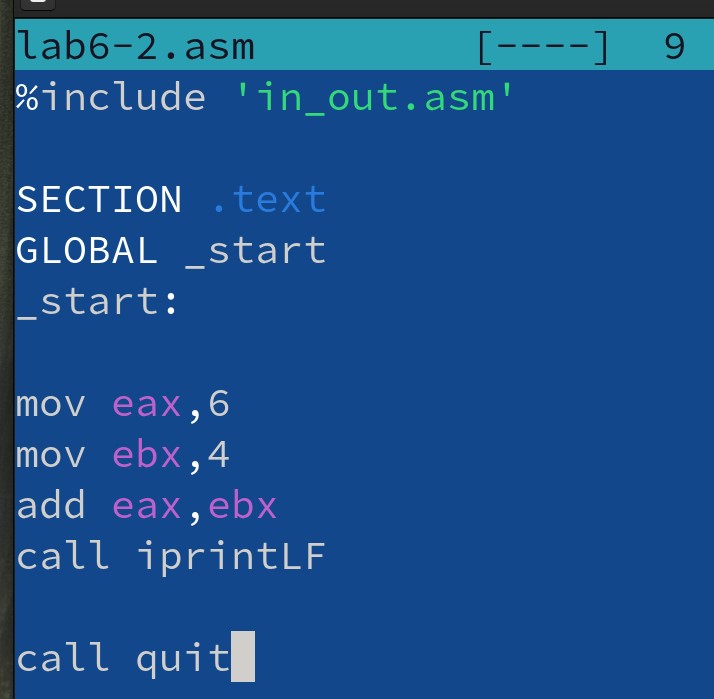


Рис. 9: Измененный lab6-2.asm

1. Создаем исполняемый файл и запускаем его (Рис. 10).

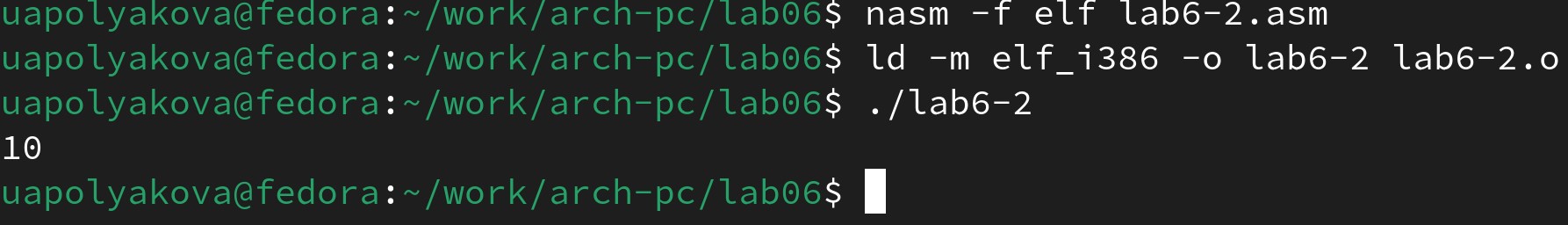


Рис. 10: Запукс измененного lab6-2.asm

В результате выведется число 10, а не символ, кодом которого является это число.

1. Заменяем iprintLF на iprint, создаем исполняемый файл и запускаем его (Рис. 11).

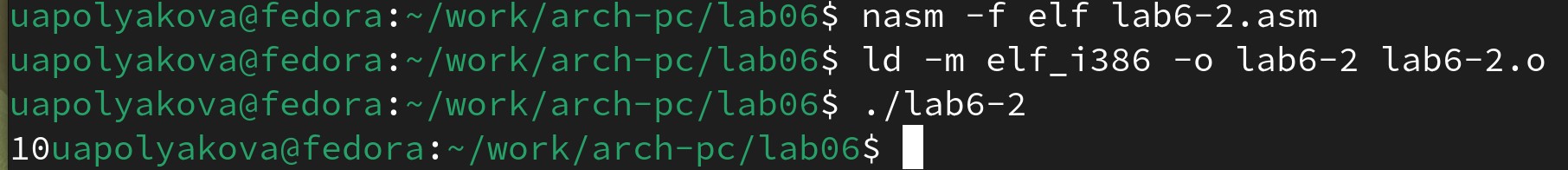


Рис. 11: Работа файла с iprint

Функция iprint выводит без переноса строки.

1. Создаем lab6-3.asm по листингу 6.3. (Рис. 12).

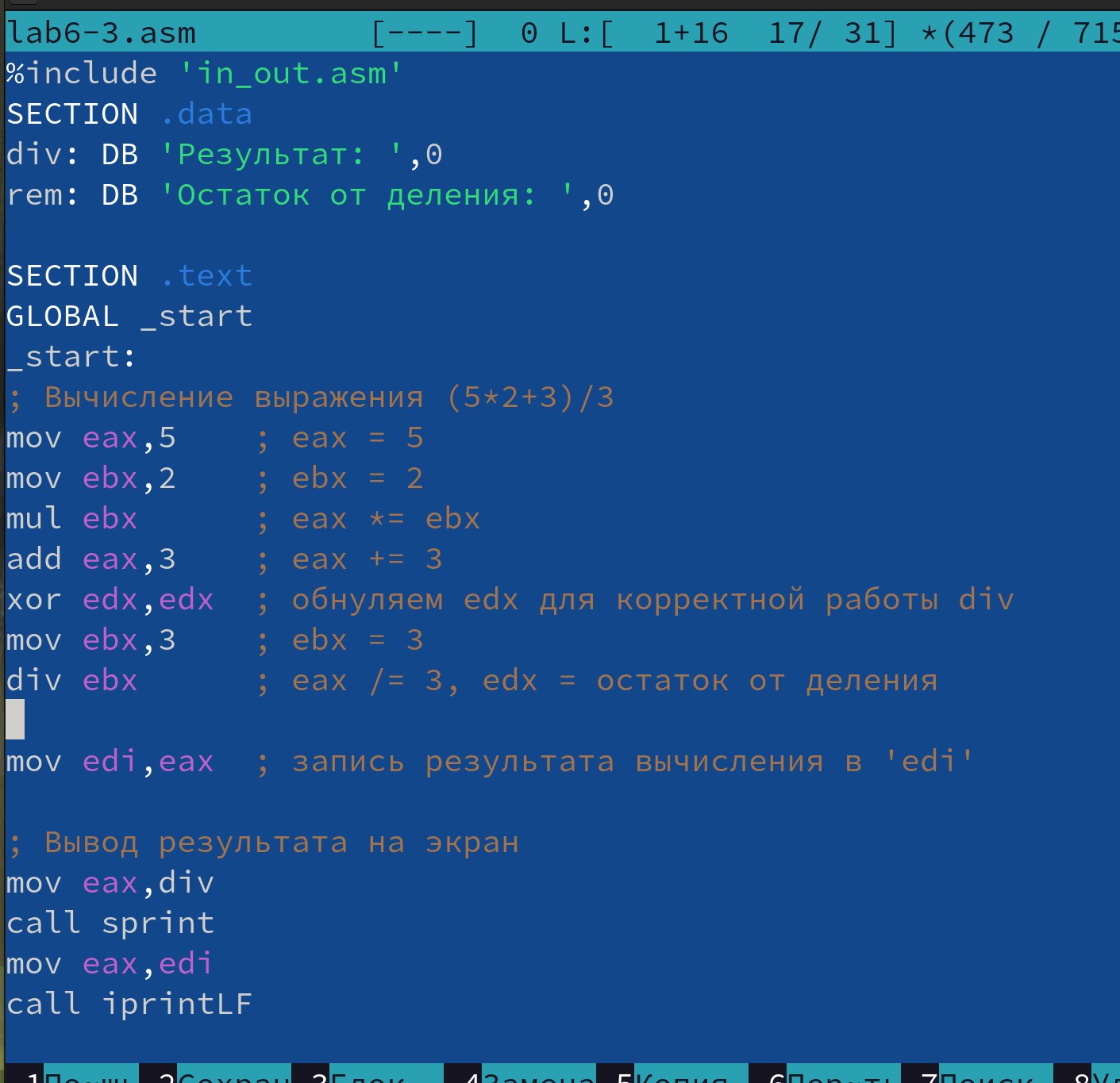


Рис. 12: lab6-3.asm из листинга 6.3.

1. Создаем исполняемый файл и запускаем его (Рис. 13).

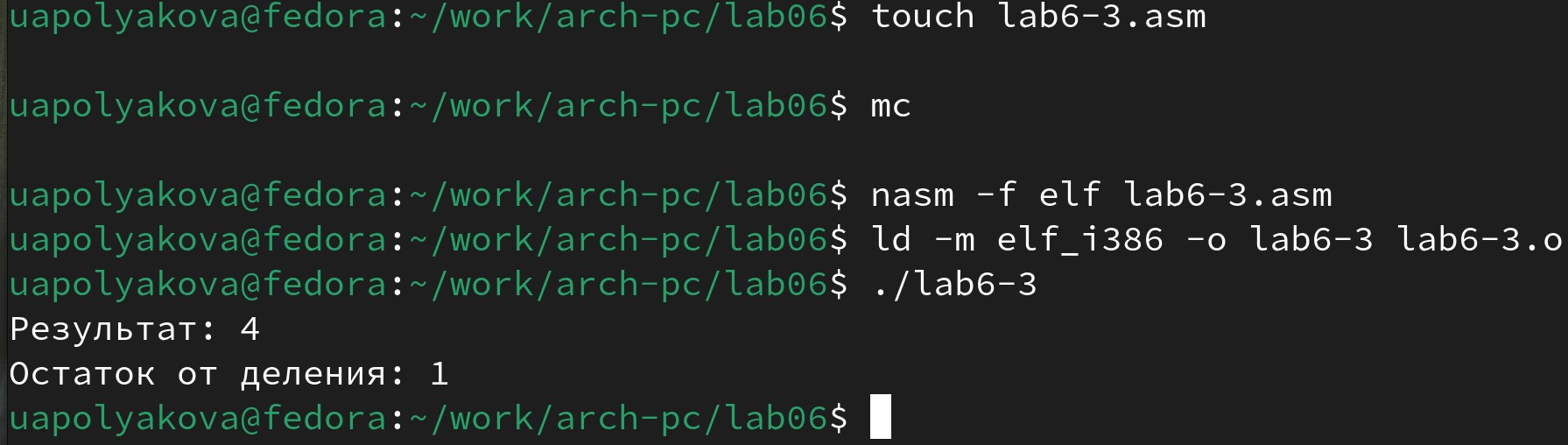


Рис. 13: Запуск lab6-3.asm из листинга 6.3.

1. Изменяем текст для вычисления (4\*6+2)/5 (Рис. 14).

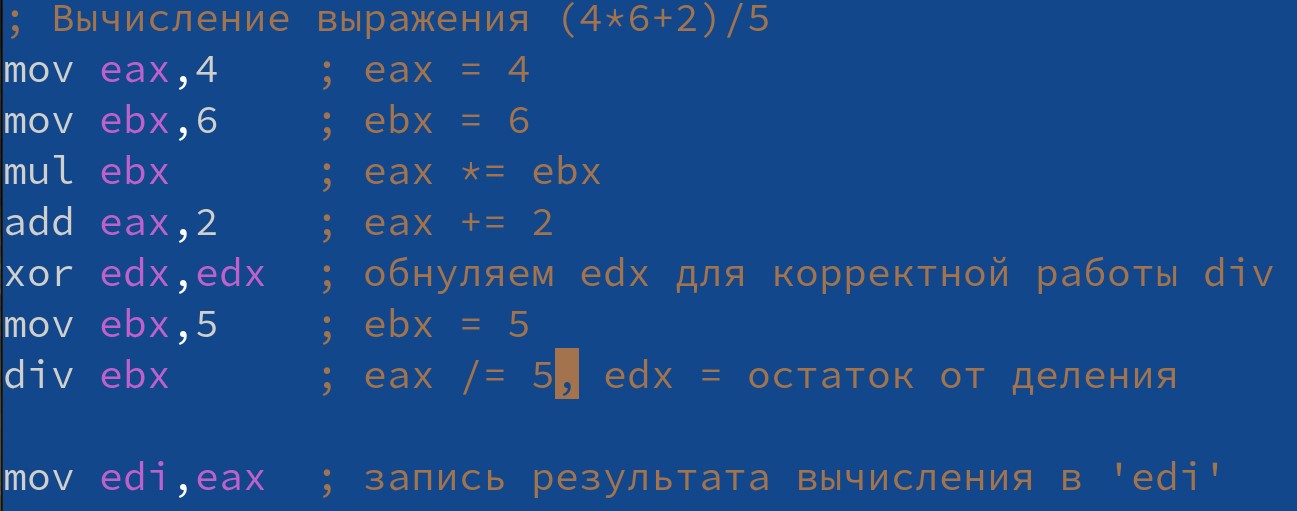


Рис. 14: Измененный lab6-3.asm

1. Создаем исполняемый файл и запускаем его (Рис. 15).

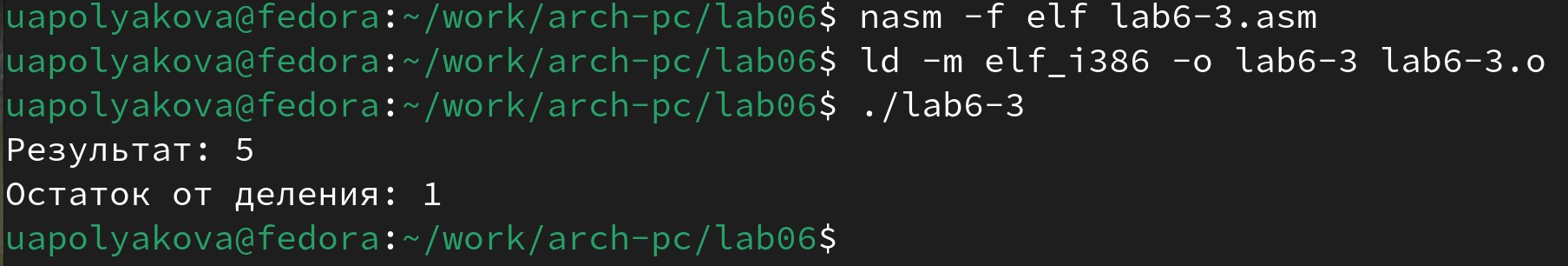


Рис. 15: Запуск измененного lab6-3.asm

1. Создаем variant.asm по листингу 6.4. (Рис. 16).

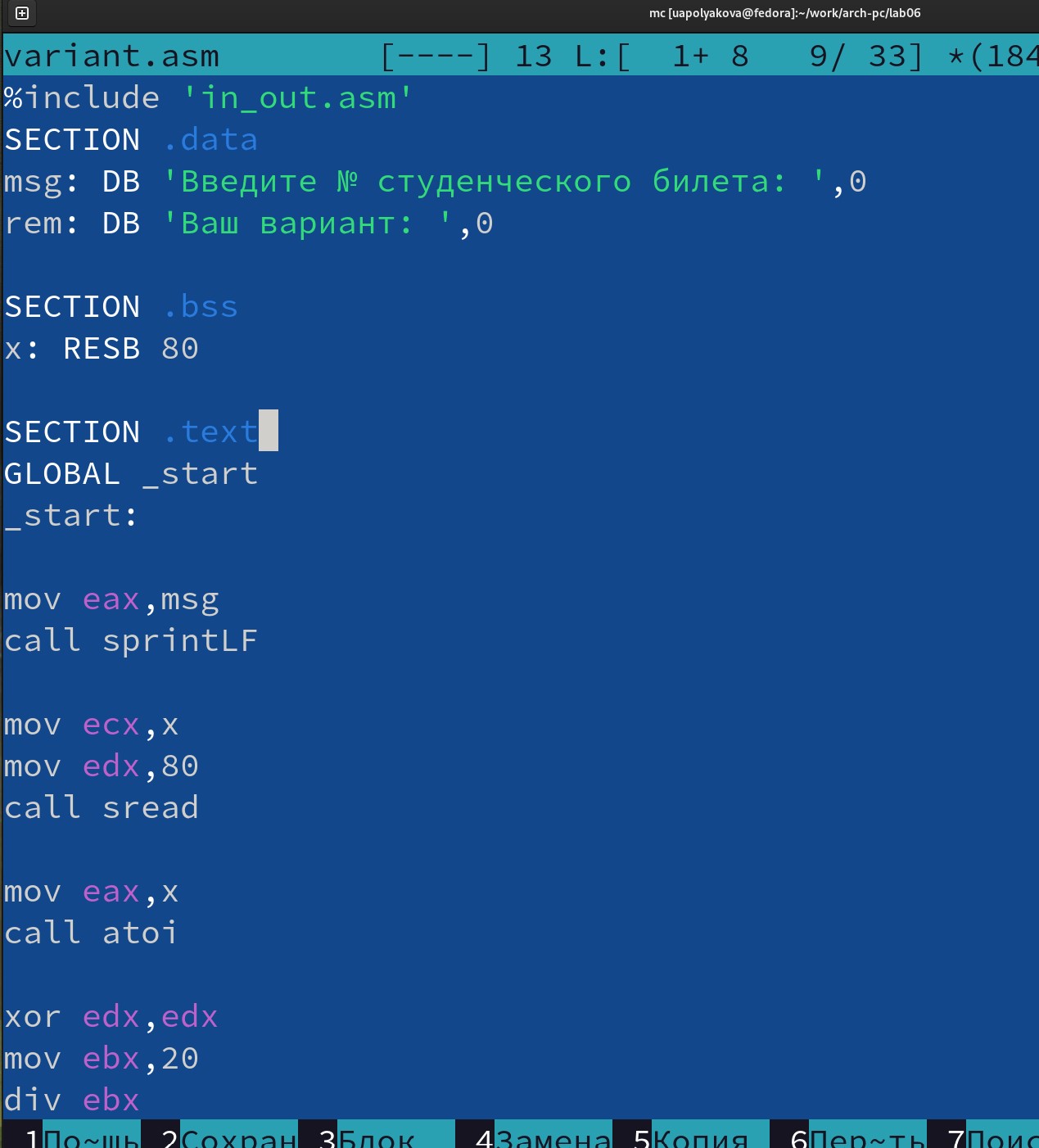


Рис. 16: variant.asm из листинга 6.4.

1. Создаем исполняемый файл и запускаем его (Рис. 17).

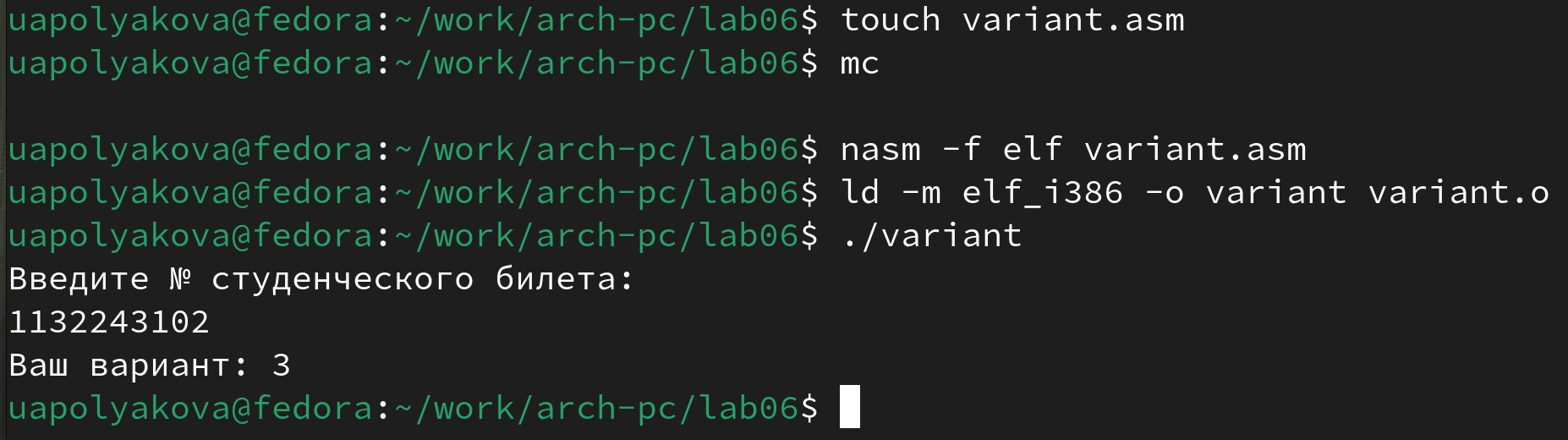


Рис. 17: Запуск variant.asm из листинга 6.4.

Проверим. Остаток от деления последних цифр 02 на 20 это 2. 2 + 1 это 3. Следовательно вариант 3.

**Ответы на вопросы:**

1. Строки

mov eax,rem  
call sprint

1. Для записи адреса переменной в ‘EAX’, записи длины вводимого сообщения в ‘EBX’ и вызова подпрограммы ввода сообщения.
2. Для преобразования символов ASCII кода в числа.
3. Строки

xor edx,edx  
mov ebx,20  
div ebx  
inc edx

1. В регистр edx.
2. Чтобы прибавить единицу (сделать инкремент) к остатку от деления на 20.
3. Строки

mov eax,edx  
call iprintLF

# 3 Результаты выполнения заданий для самостоятельной работы

1. Вариант 3. Программа для вычисления (2 + х)^2 в файле samrab.asm (рис. 18).

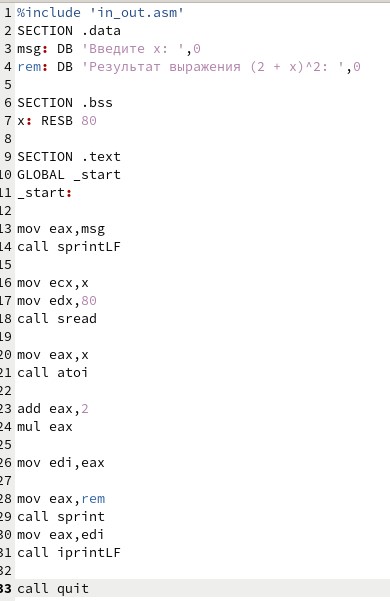


Рис. 18: Программа для вычисления (2 + х)^2

1. Создаем исполняемый файл и проверяем его работу (рис. 19).

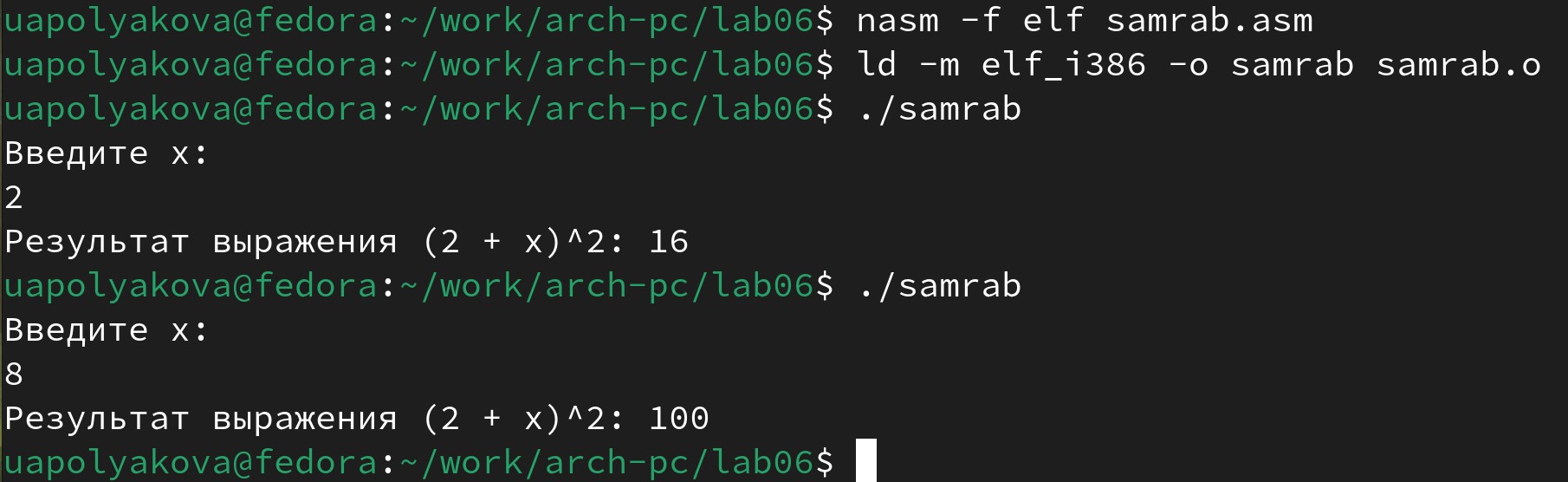


Рис. 19: Проверка работы при х1 = 2 и х2 = 8

# 4 Вывод

Были освоены арифметические инструкции языка ассемблера NASM.