**Отчет по лабораторной работе №12**



**Тема:**

**Программирование в командном процессоре ОС UNIX. Ветвления и циклы**

**Российский Университет Дружбы Народов**

**Факультет Физико-Математических и Естественных Наук**

*Дисциплина: Операционные системы*

Студент: Мухамедияр Адиль

Группа: НКНбд-01-20

Москва, 2021г.

**Цель работы**

Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций ициклов.

**Введение**

Операции с блоками кода — ключ к структурированным и упорядоченным сценариям оболочки. Циклы и ветвления являются теми инструментальными средствами,которые предоставляют возможность достигнуть этой цели.

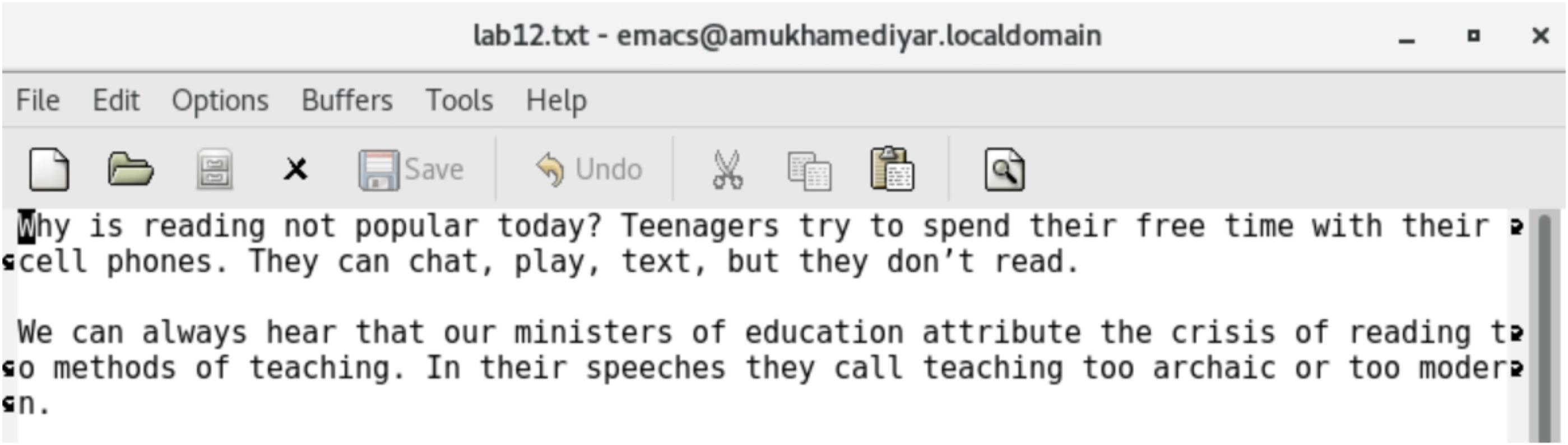
**Циклы**

Цикл это блок команд, который повторяется (итерируется) до тех пор, пока не будет выполнено условие выхода из цикла .

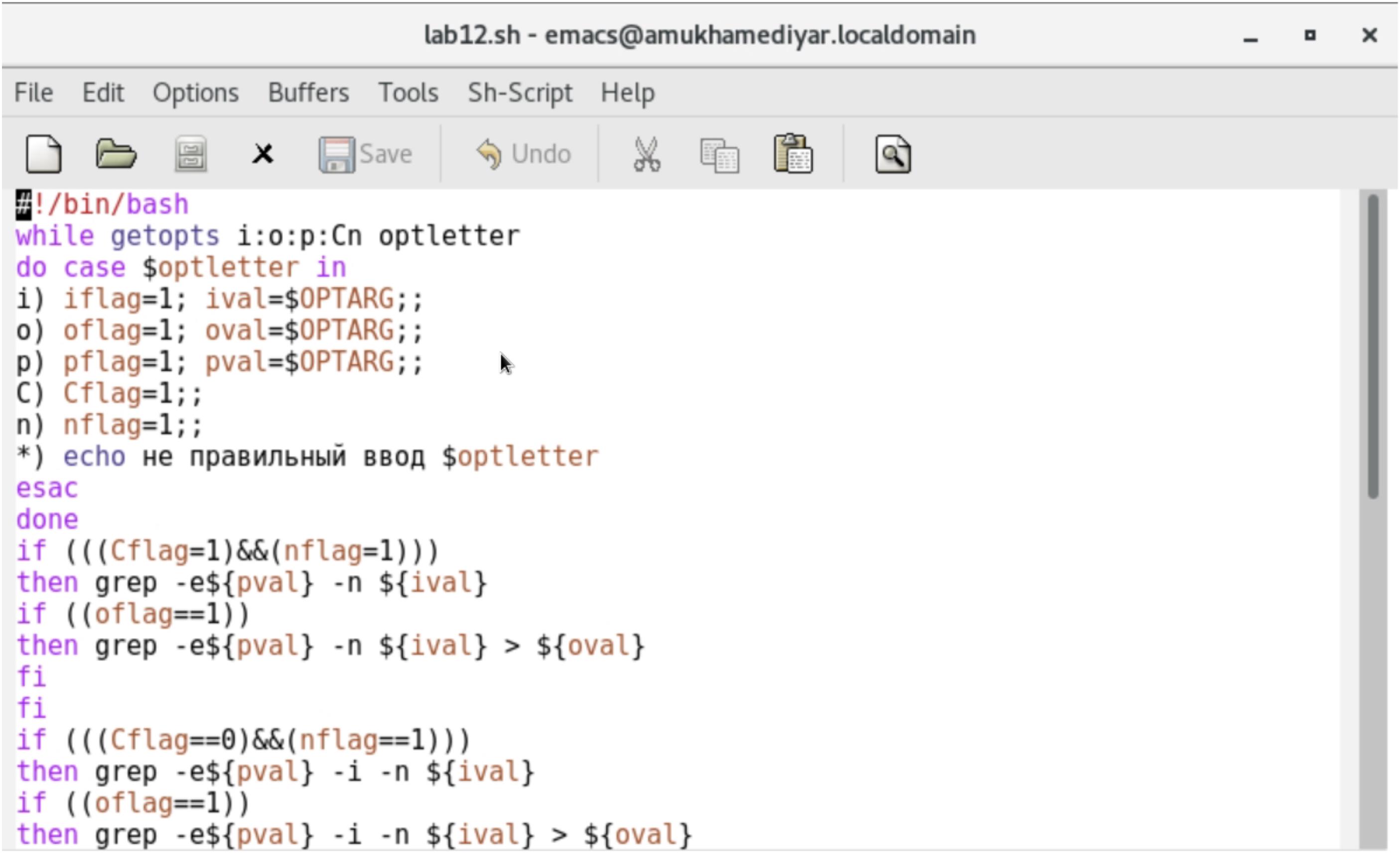
**Ход работы:**

1. Для начало я скопировал любой текст из интернета.

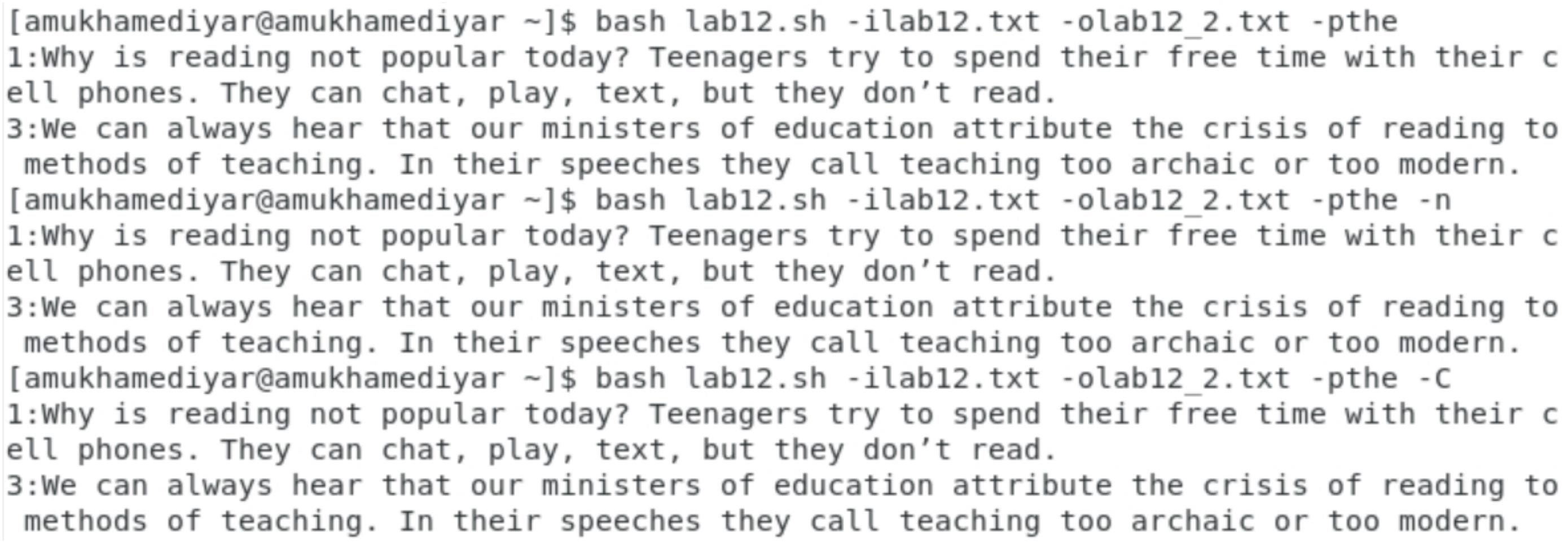
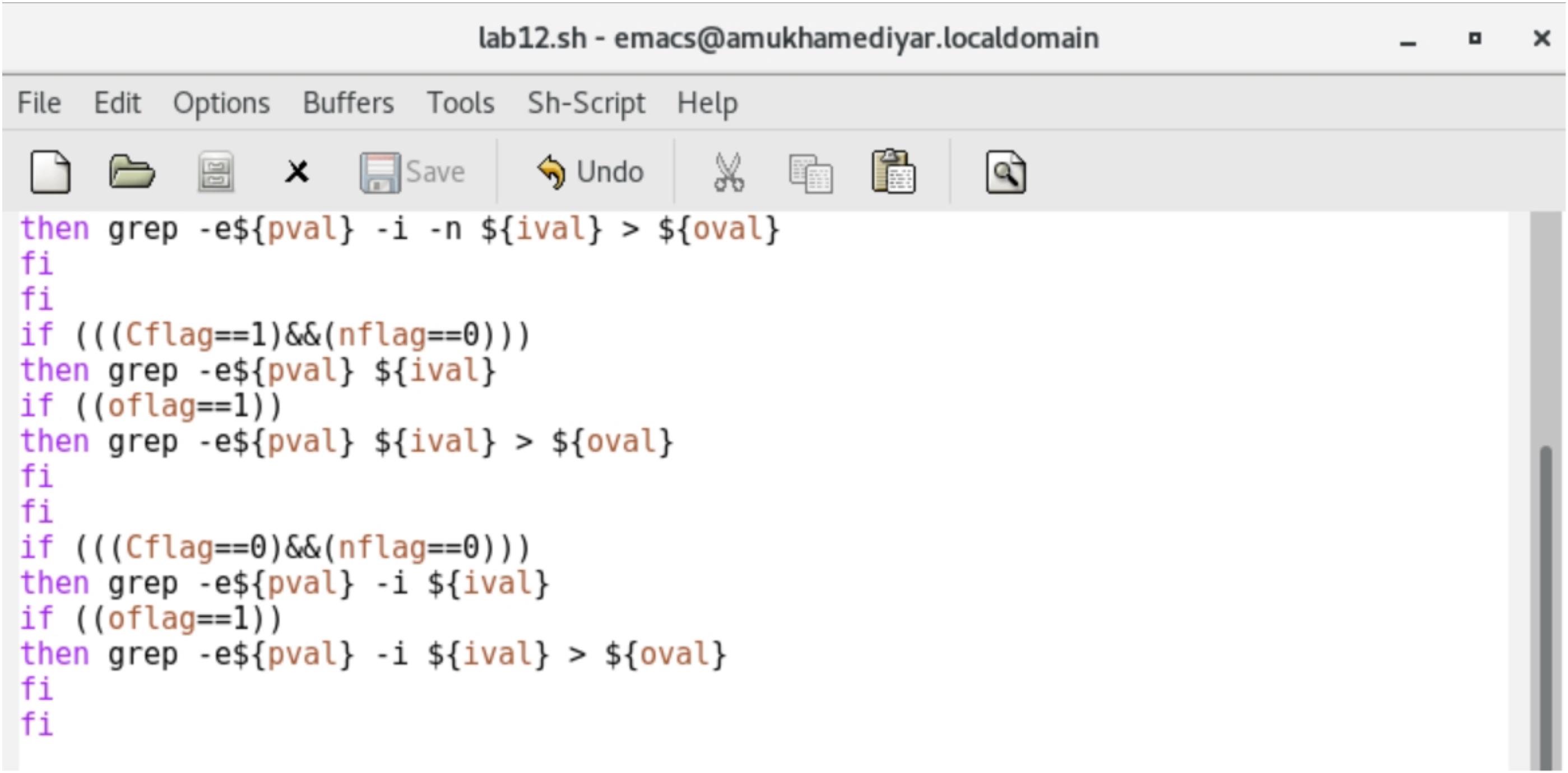
• Используя команды getopts grep, написал командный файл.



• Данный командный файл анализирует командную строку с ключами: -iinputfile — прочитать данные из указанного файла; -ooutputfile — вывести данные в указанный файл; -pшаблон — указать шаблон для поиска; -C — различать большие и малые буквы; -n — выдавать номера строк.

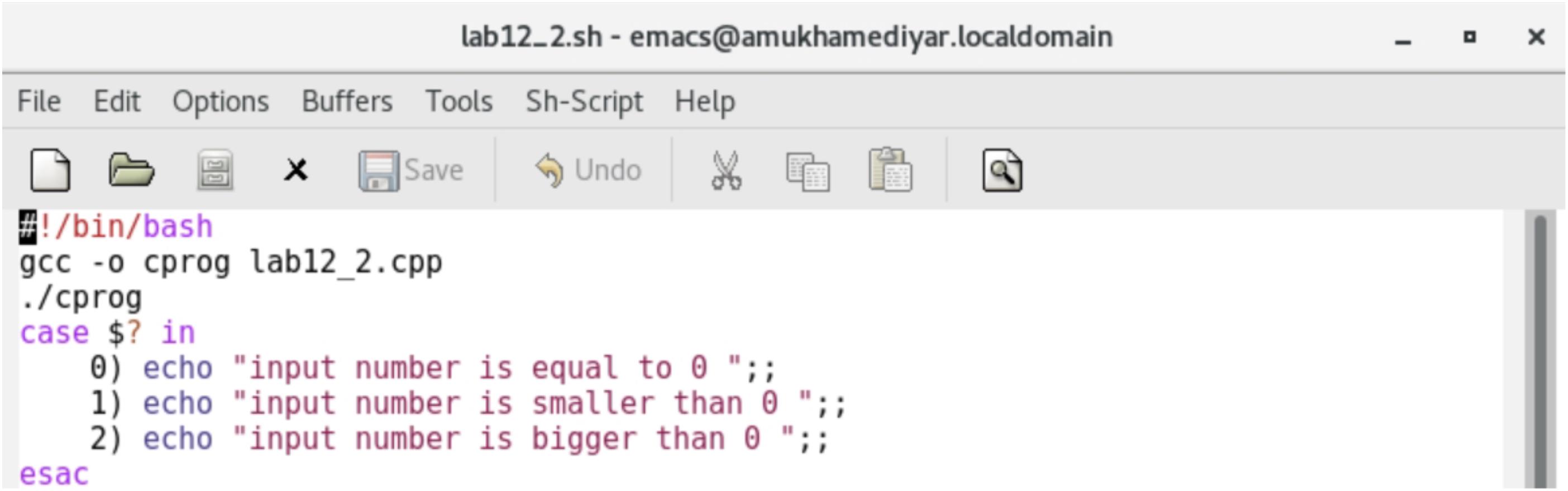
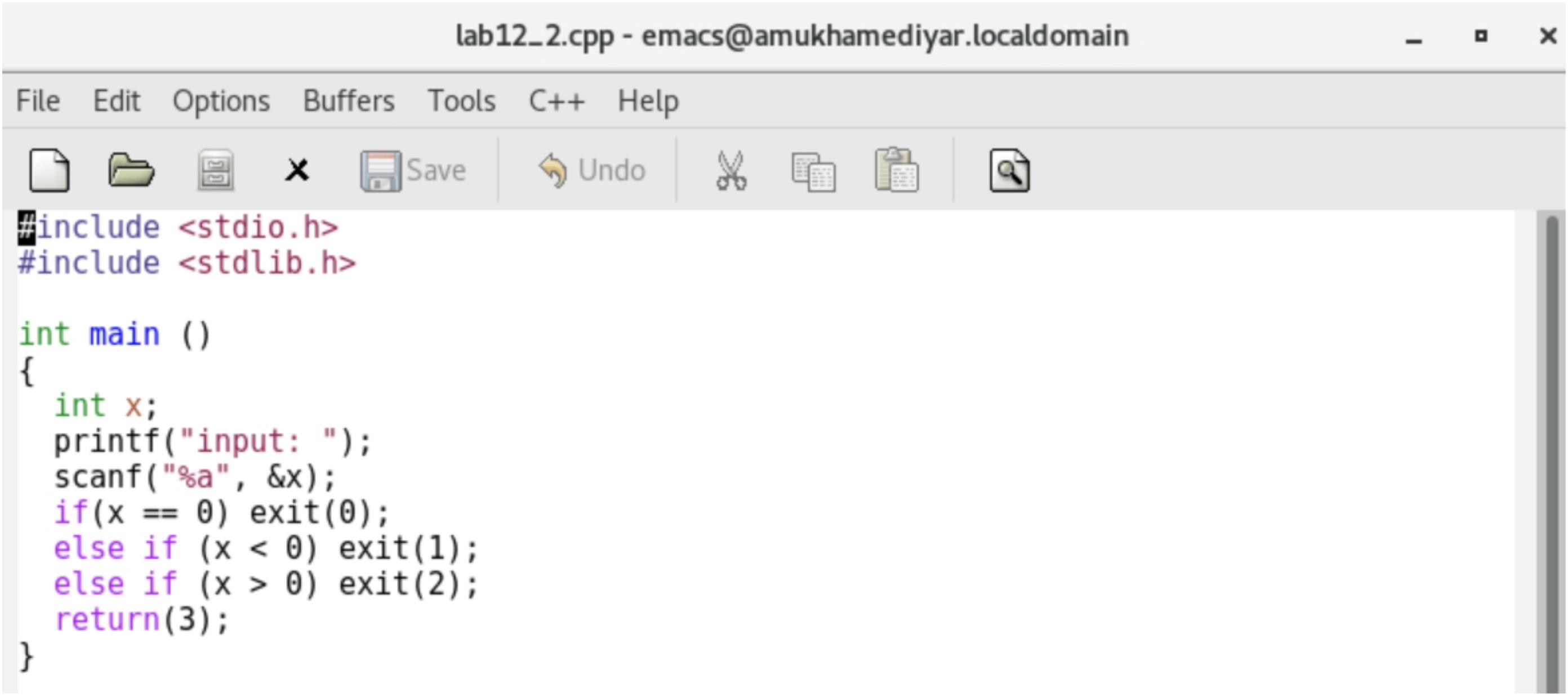


• После я написал слово, которую программа ищет в моем тексте, используя ключ -p.



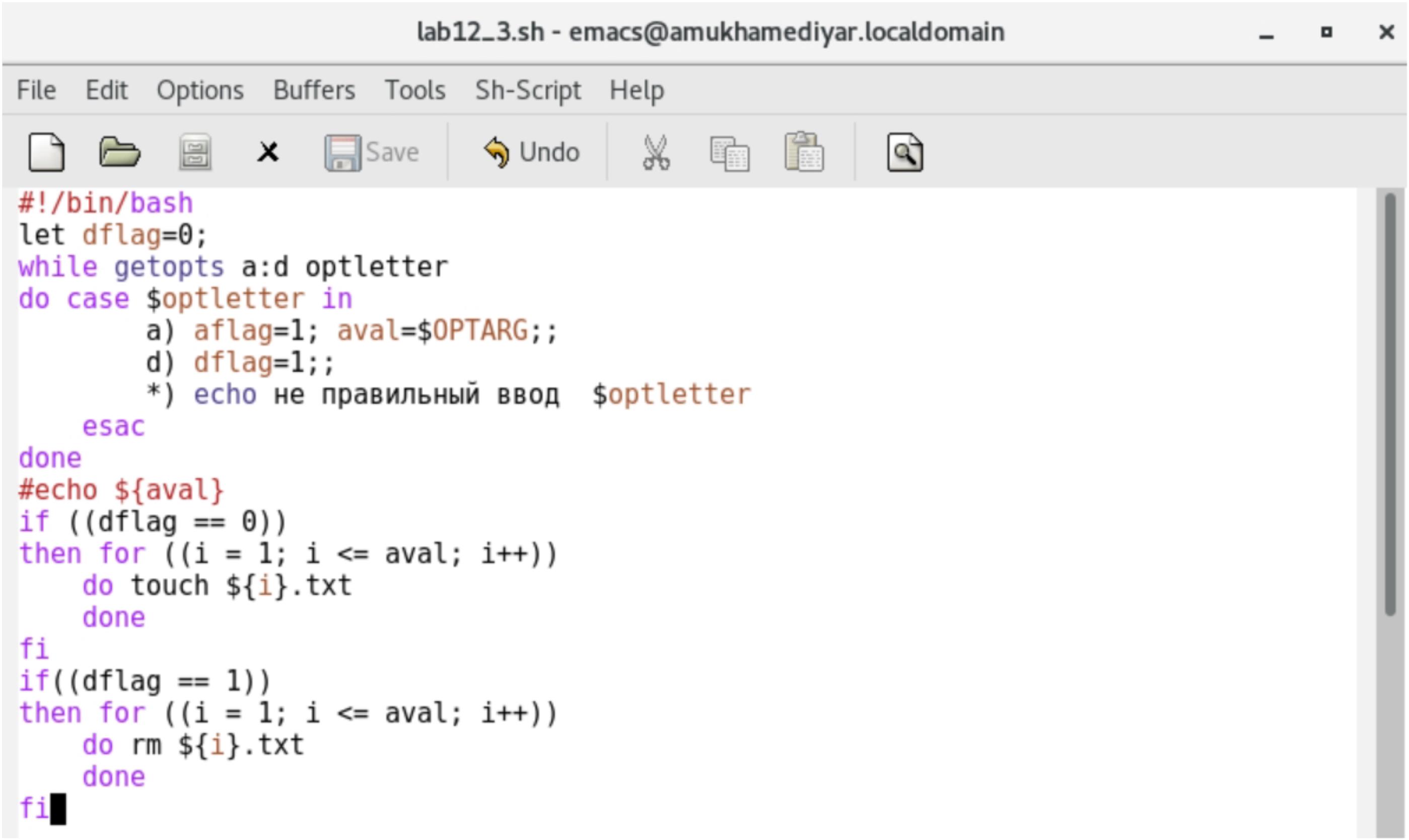
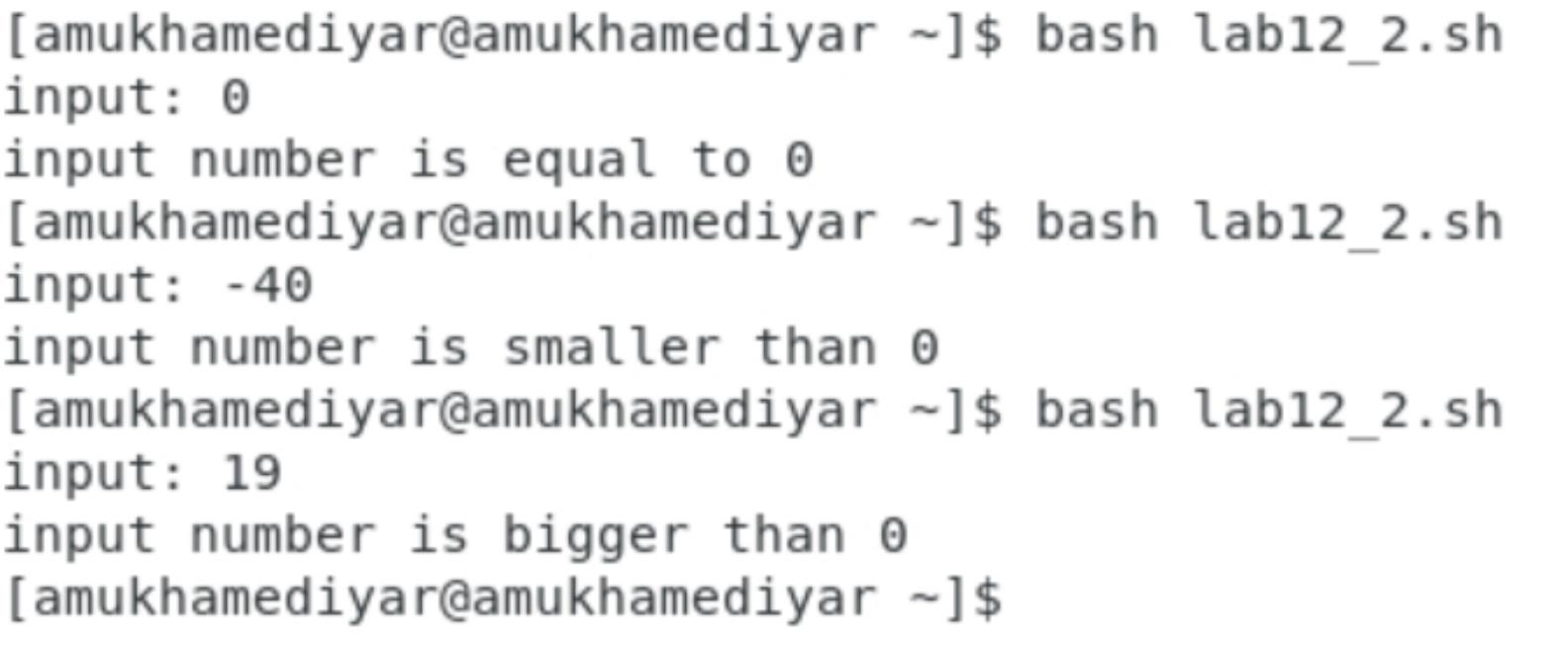
2. Написал на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции exit(n), передавая информацию о коде завершения в оболочку.

• Командный файл вызывает эту программу и, проанализировав с помощью команды $?, выдает сообщение о том, какое число было введено.



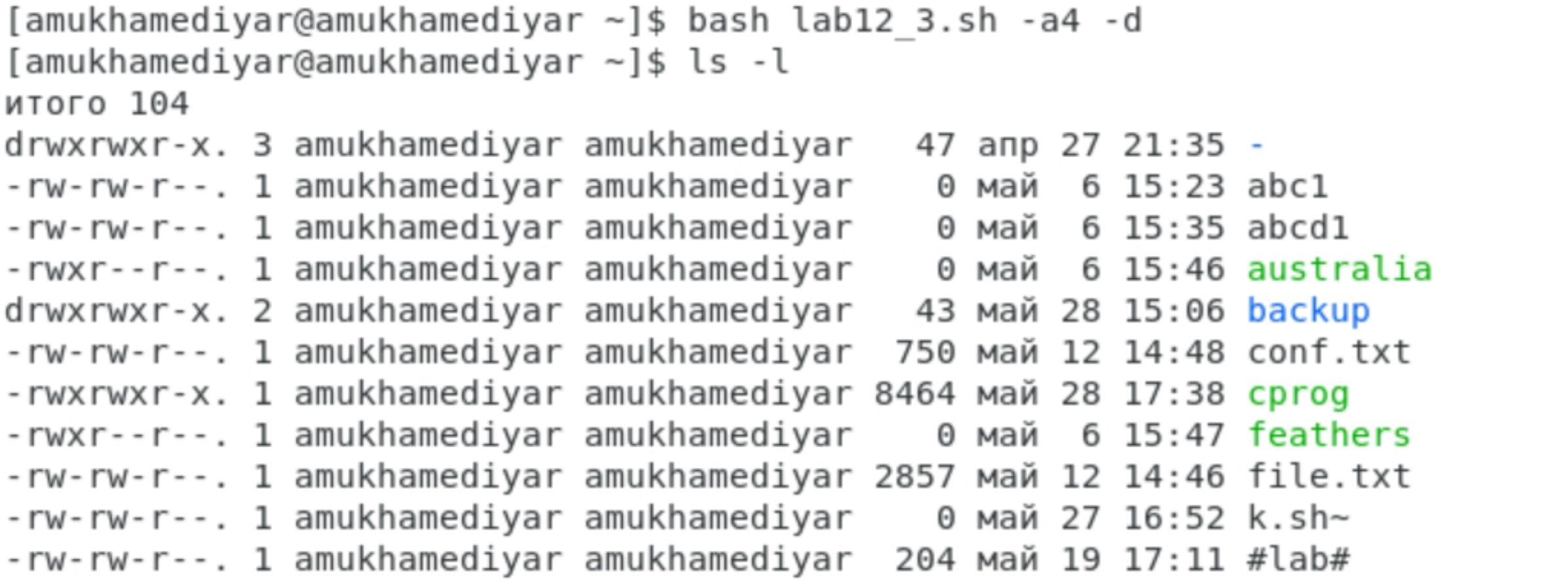
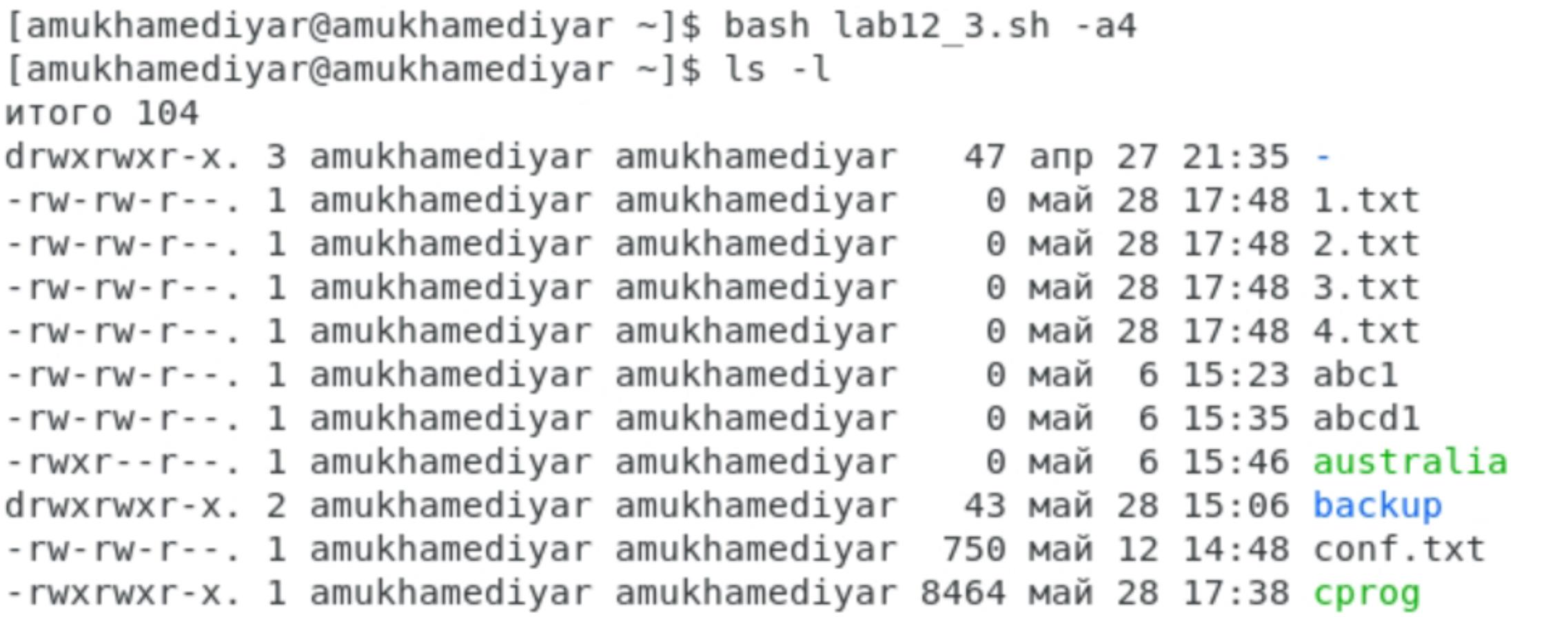
• Проверил свой код.

3. Написал командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.).



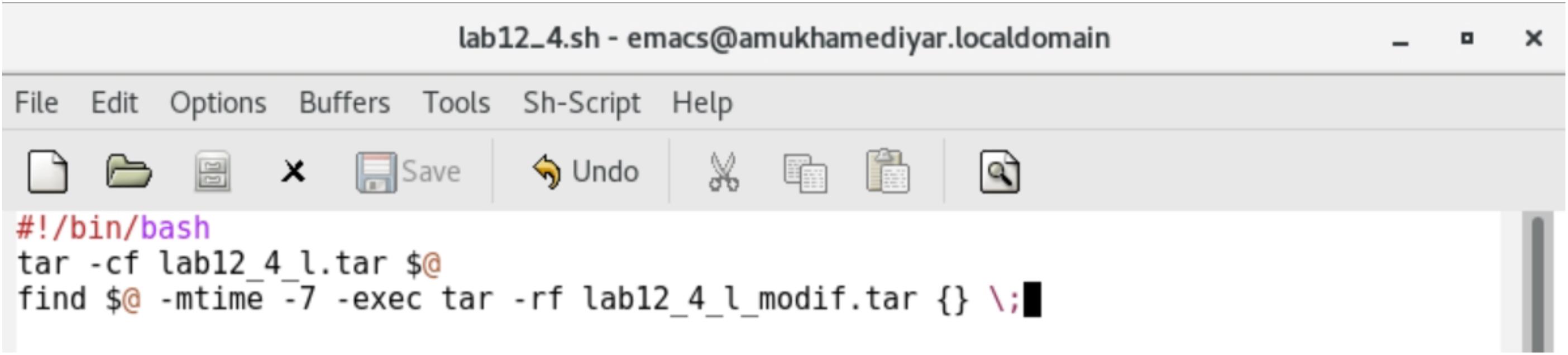
• Число файлов, которые необходимо создать, передаётся в аргументы командной строки.

• Этот же командный файл удаляет все созданные им файлы (если они существуют).



4. Написал командный файл, который с помощью команды*tar* запаковывает в архив все файлы в указанной директории.

• Модифицировал его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовал команду find).



**Вывод**

Изучил основы программирования в оболочке ОС UNIX, научился писать более сложные командные файлы с использованием логических управляющих конструкций ициклов.

**Библиография**

[Домашний склерозник](https://www.skleroznik.in.ua/2013/07/31/cikly-i-vetvleniya/)

**Ответы на контрольные вопросы:**

1. Команда getopts является встроенной командой командной оболочки bash, предназначенной для разбора параметров сценариев. Она обрабатывает исключительно однобуквенные параметры как с аргументами, так и без них и этого вполне достаточно для передачи сценариям любых входных данных.

2. При генерации имен используют метасимволы:

• произвольная (возможно пустая) последовательность символов; ? один произвольный символ; [...] любой из символов, указанных в скобках перечислением и/или с указанием диапазона; cat f\* выдаст все файлы каталога, начинающиеся с "f"; cat *f* выдаст все файлы, содержащие "f"; cat program.? выдаст файлы данного каталога с однобуквенными расширениями, скажем "program.c" и "program.o", но не выдаст "program.com"; cat [a-d]\* выдаст файлы, которые начинаются с "a", "b", "c", "d". Аналогичный эффект дадут и команды "cat [abcd]*" и "cat [bdac]*".

3. Операторы && и || являются управляющими операторами. Если в командной строке стоит command1 && command2, то command2 выполняется в том, и только в том случае, если статус выхода из команды command1 равен нулю, что говорит об успешном ее завершении. Аналогично, если командная строка имеет вид command1 || command2, то команда command2 выполняется тогда, и только тогда, когда статус выхода из команды command1 отличен от нуля.

4. Оператор break завершает выполнение ближайшего включающего цикла или условного оператора, в котором он отображается.

5. Команда true всегда возвращает ноль в качестве выходного статуса для индикации успеха. Команда false всегда возвращает не-ноль в качестве выходного статуса для индикации неудачи. Во всех управляющих конструкциях в качестве логического значения используется код возврата из программы, указанной в качестве условия. Код возврата 0 – истина, любое другое значение – ложь. Программа true – всегда завершается с кодом 0, false – всегда завершается с кодом 1.

6. Введенная строка означает условие существования файла man$s/$i.$s

7. Цикл While выполняется до тех пор, пока указанное в нем условие истинно. Когда указанное условие становится ложным - цикл завершается. Цикл Until выполняется до тех пор, пока указанное в нем условие ложно.