# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

### Факультет физико-математических и естественных наук Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №11 ===============

дисциплина: Операционные системы

Студент: Койфман Кирилл Дмитриевич

Группа: НПИбд-01-21

## Введение.

## Цель работы.

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

## Задачи.

1. Используя команды getopts grep, написать командный файл, который 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.

а затем ищет в указанном файле нужные строки, определяемые ключом -p.

1. Написать 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 в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено.
2. Написать 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 должен уметь удалять все созданные им файлы (если они существуют).
3. Написать командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find).

## Ход работы.

## 1 задание.

Используя команды getopts grep, напишем командный файл, который анализирует командную строку с ключами:

– -iinputfile — прочитать данные из указанного файла;

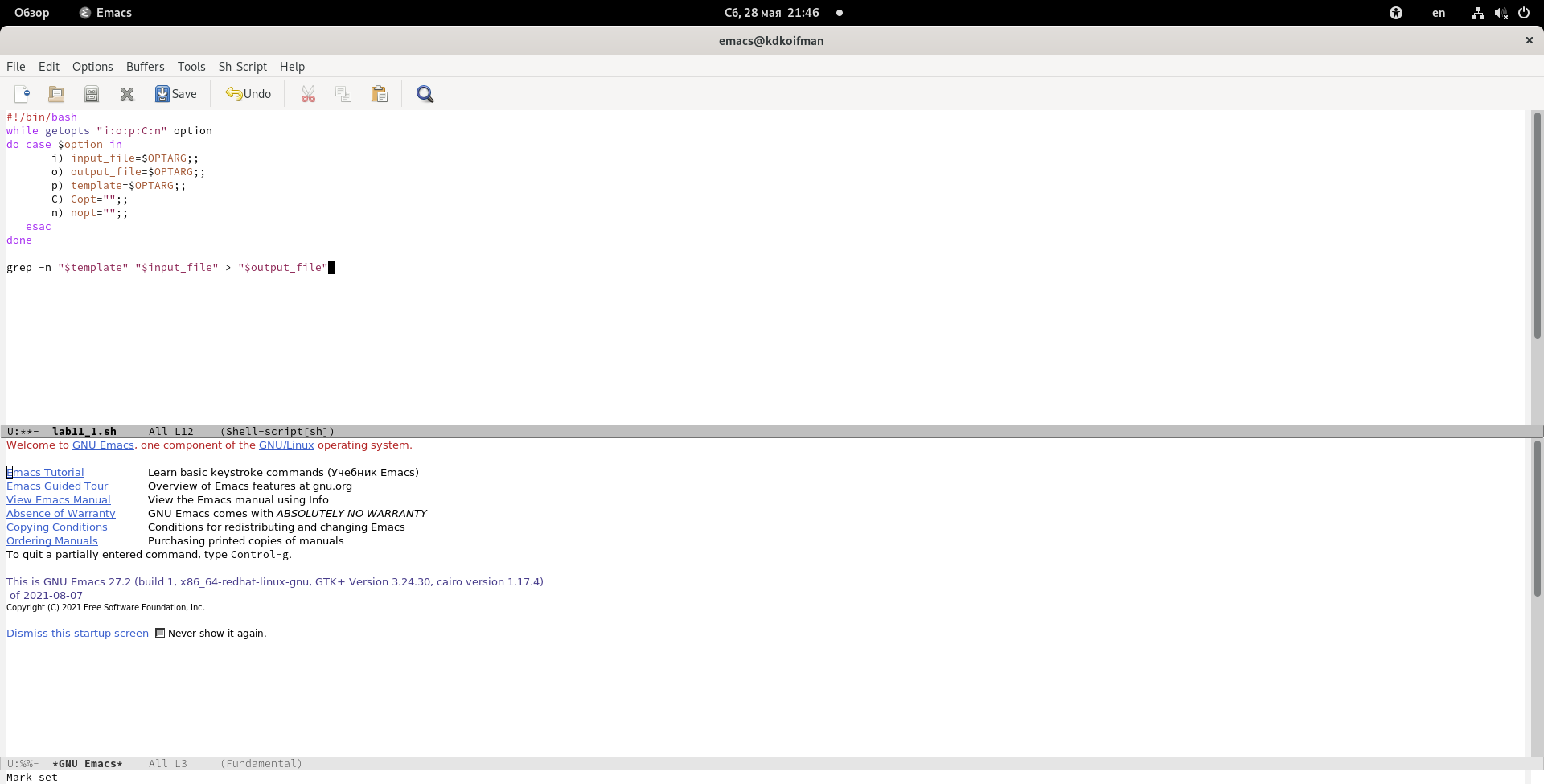
– -ooutputfile — вывести данные в указанный файл;

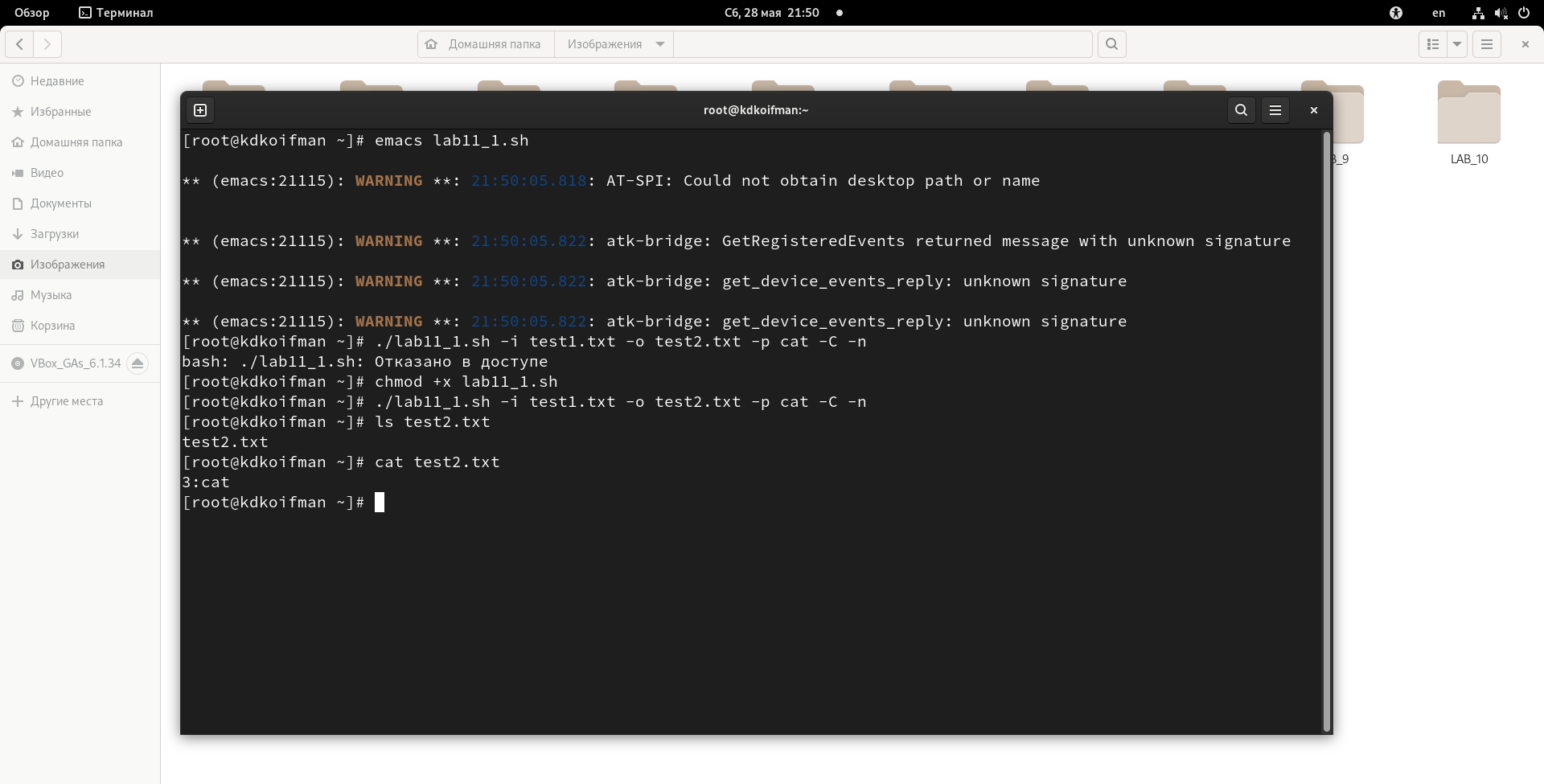
– -pшаблон — указать шаблон для поиска;

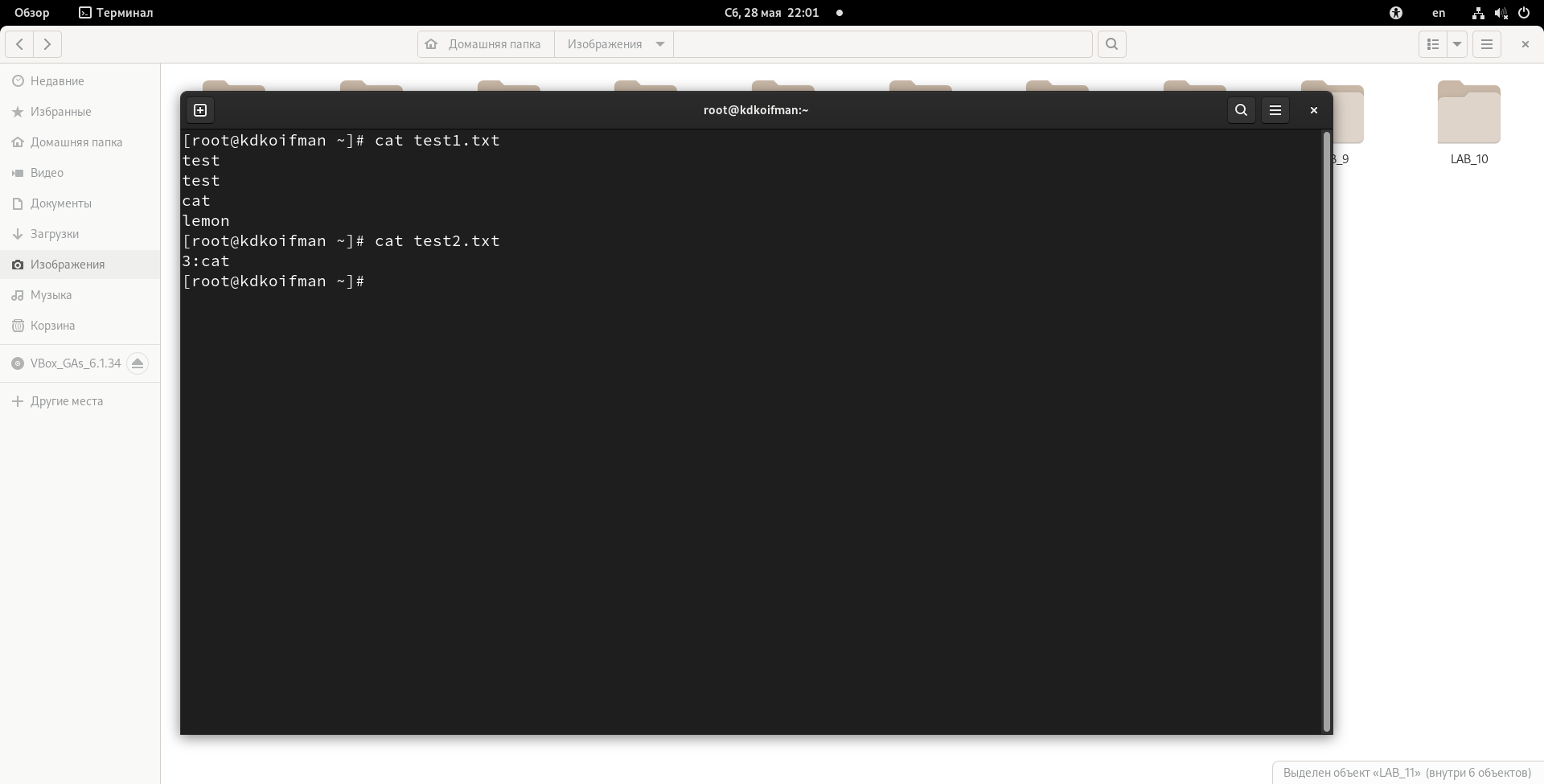
– -C — различать большие и малые буквы;

– -n — выдавать номера строк.

а затем ищет в указанном файле нужные строки, определяемые ключом -p (рис.1, 2, 3)

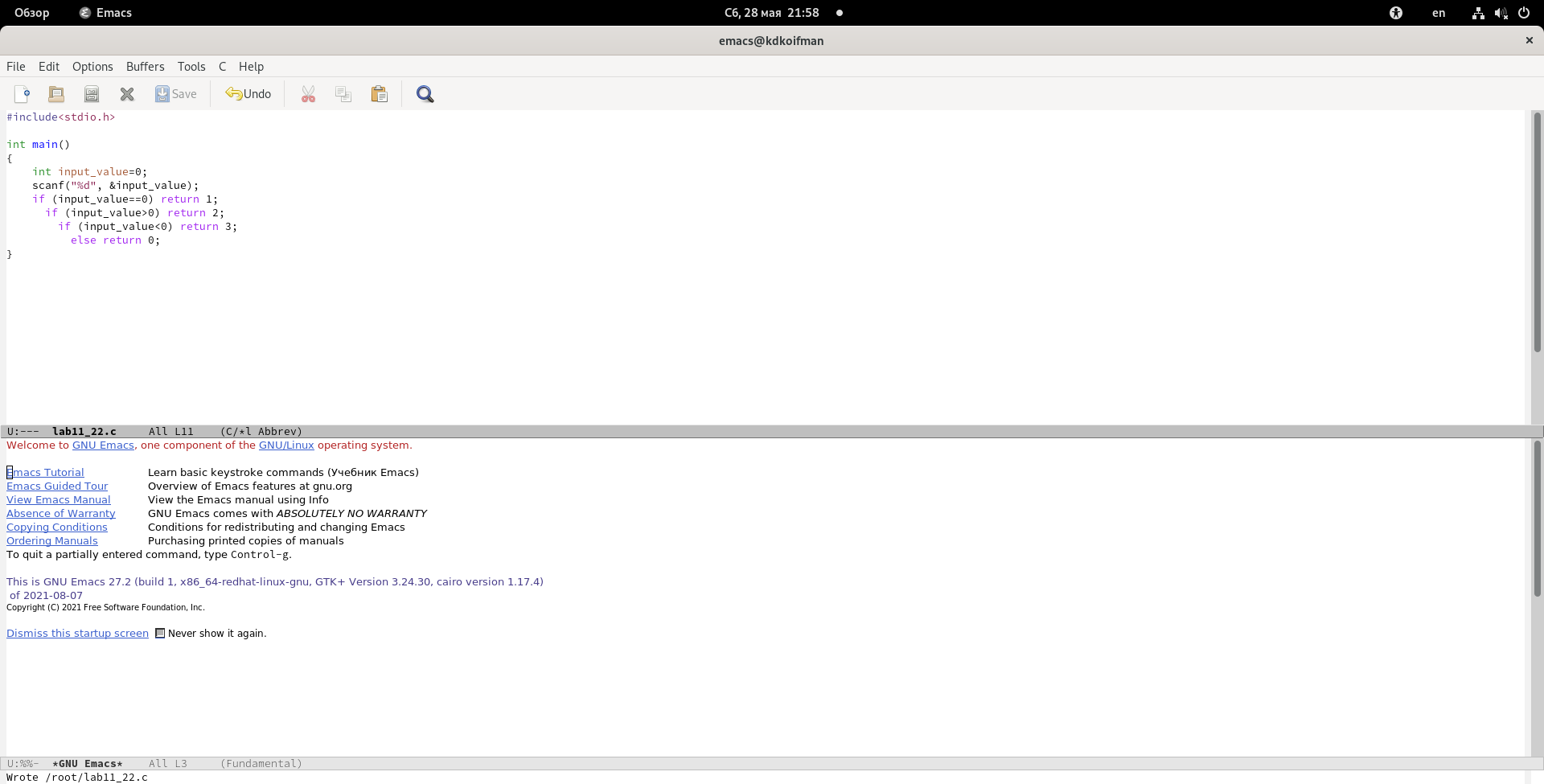
 :–: *рис.1(код командного файла)*

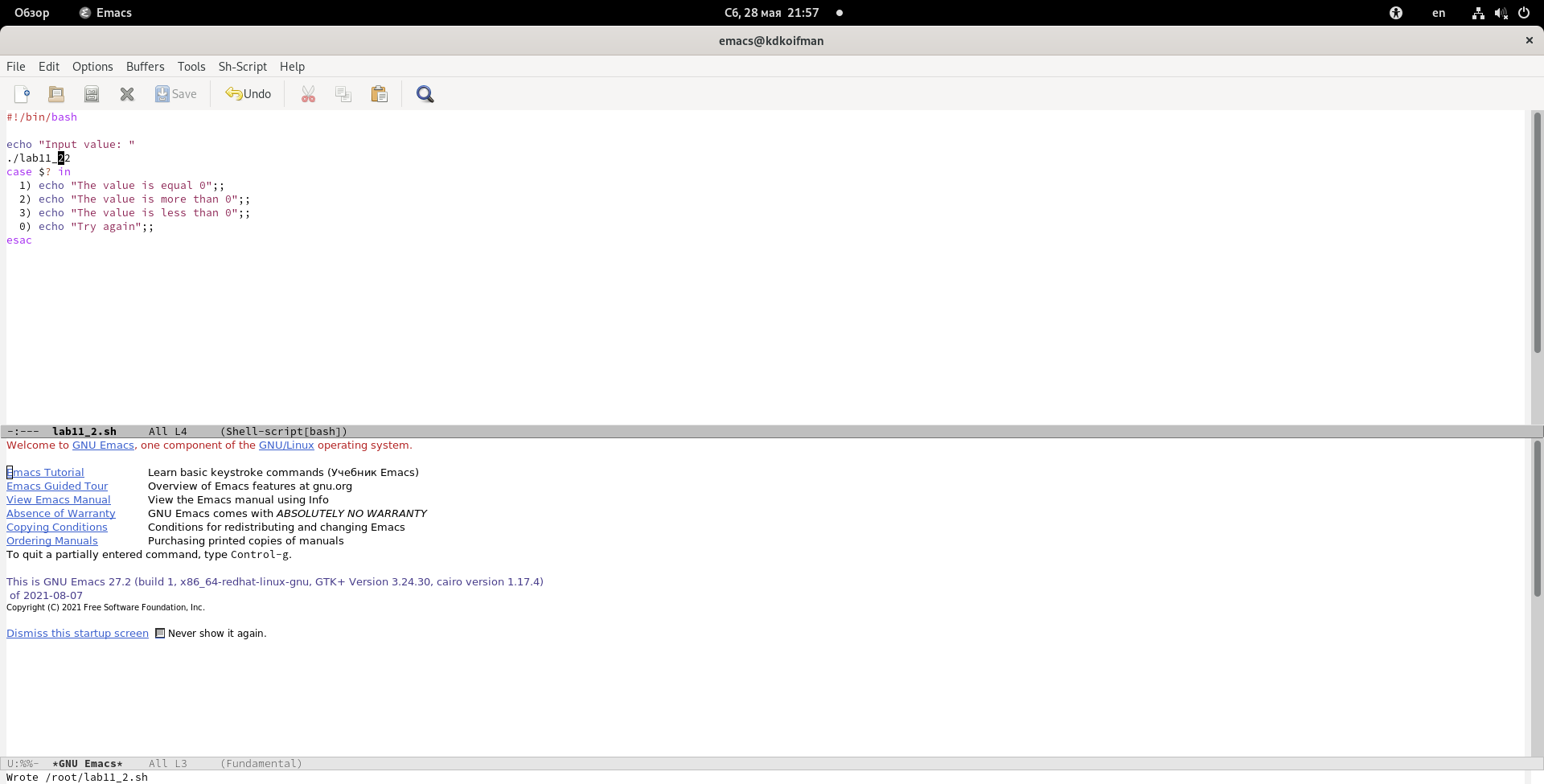
 :–: *рис.2(выполнение командного файла)*

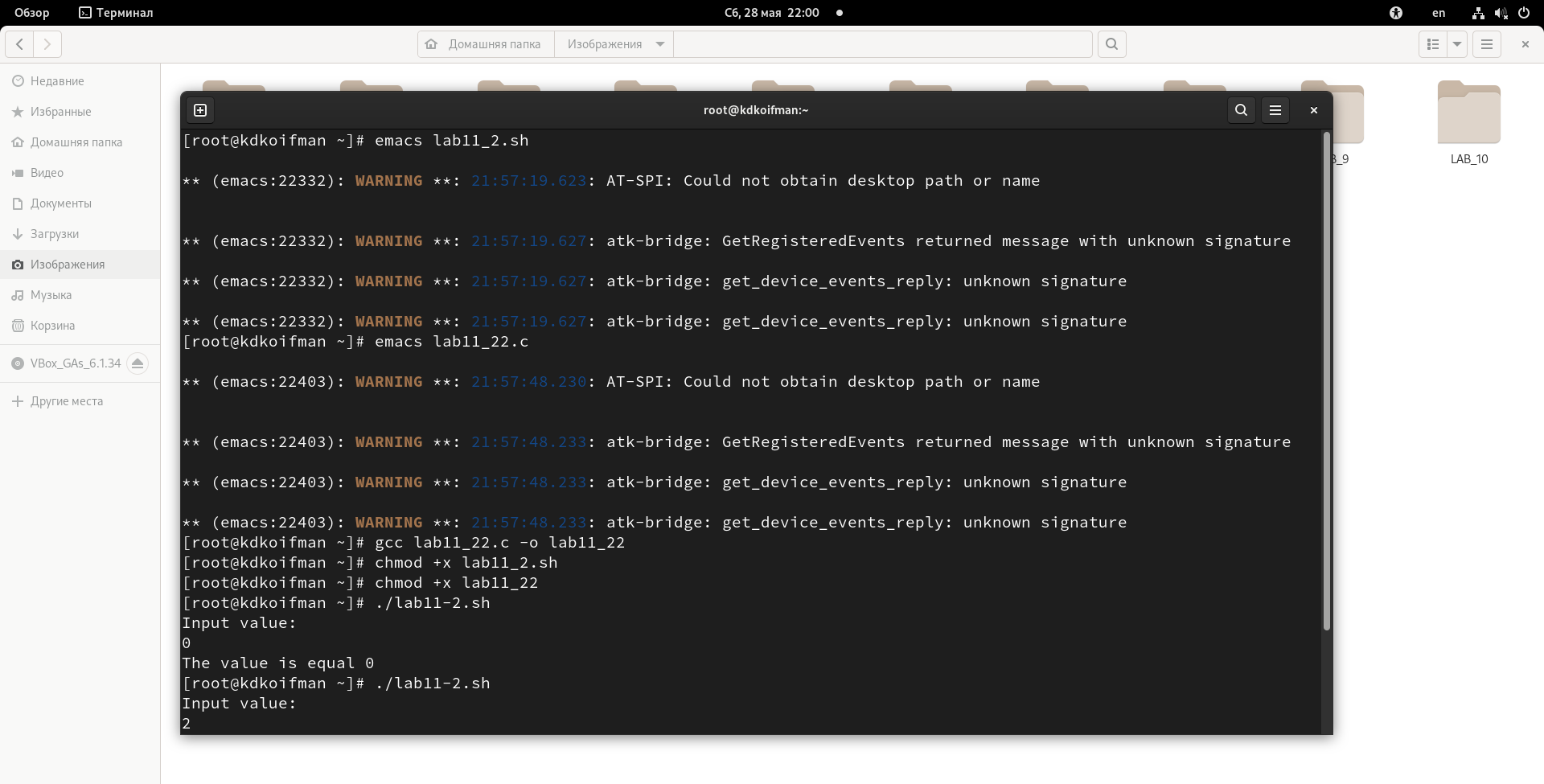
 :–: *рис.3(содержимое файлов test1.txt и test2.txt)*

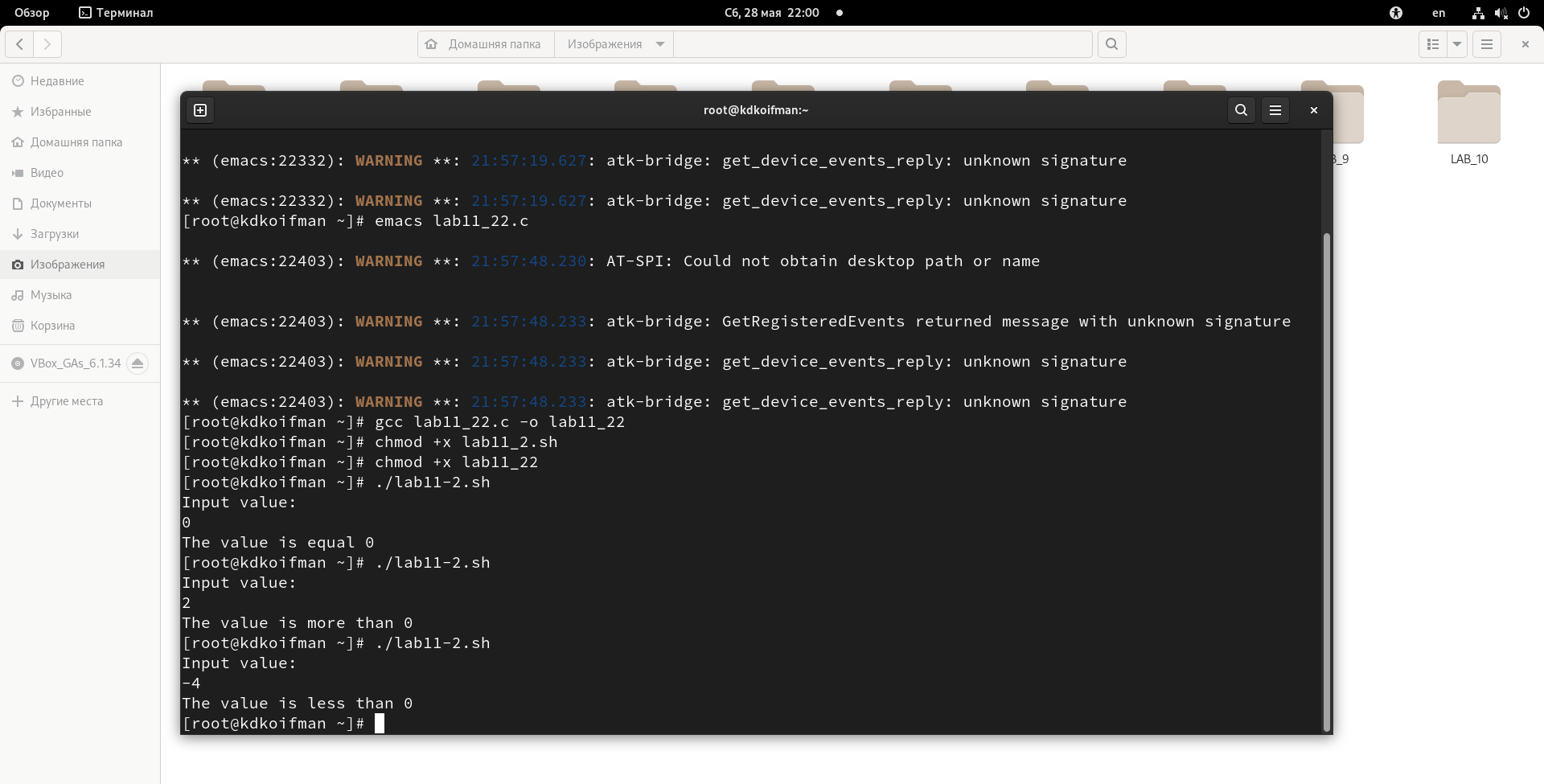
## 2 задание.

Напишем 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 в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено (рис.4, 5, 6, 7)

 :–: *рис.4(код из С-файла)*

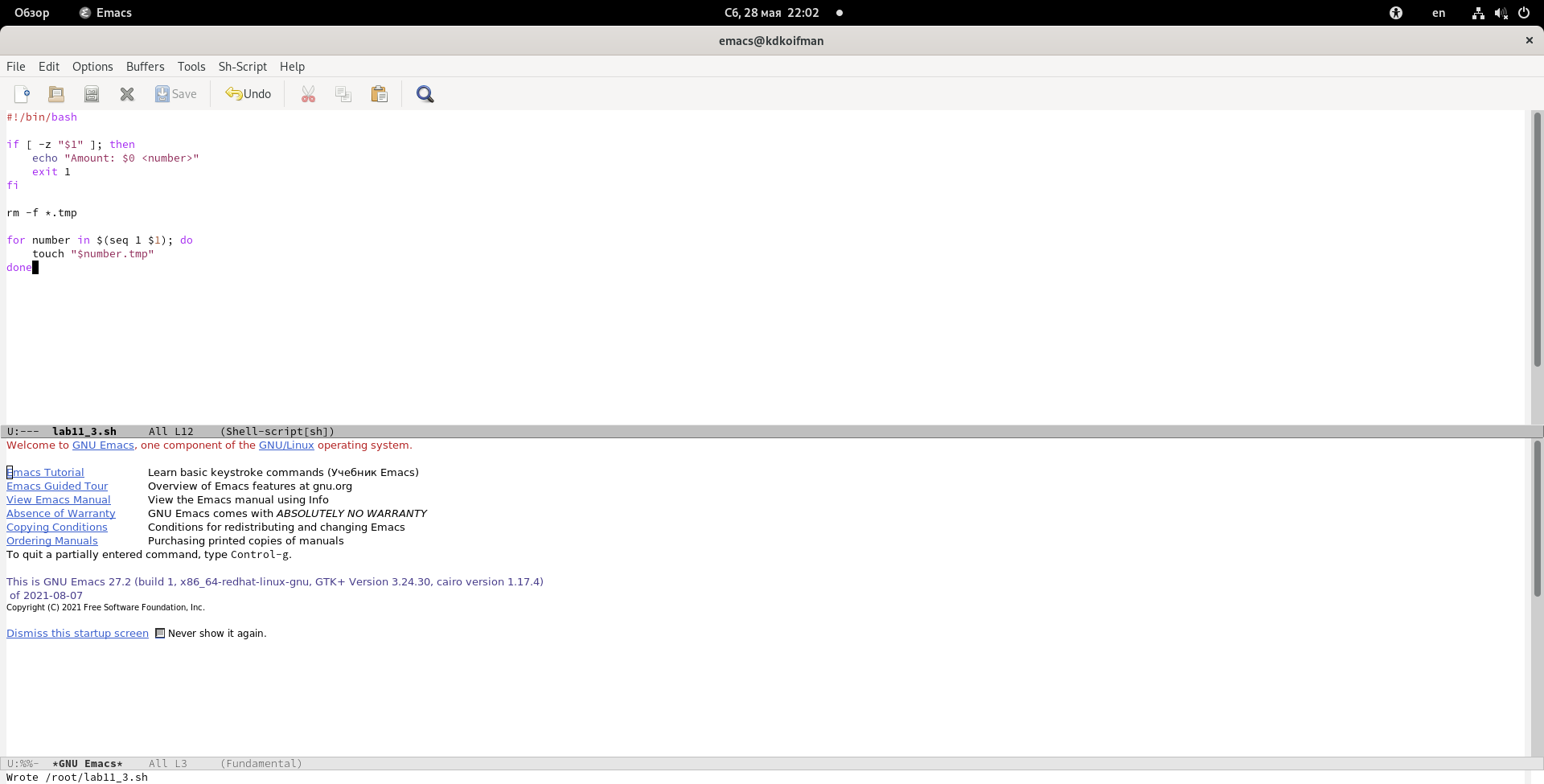
 :–: *рис.5(код командного файла)*

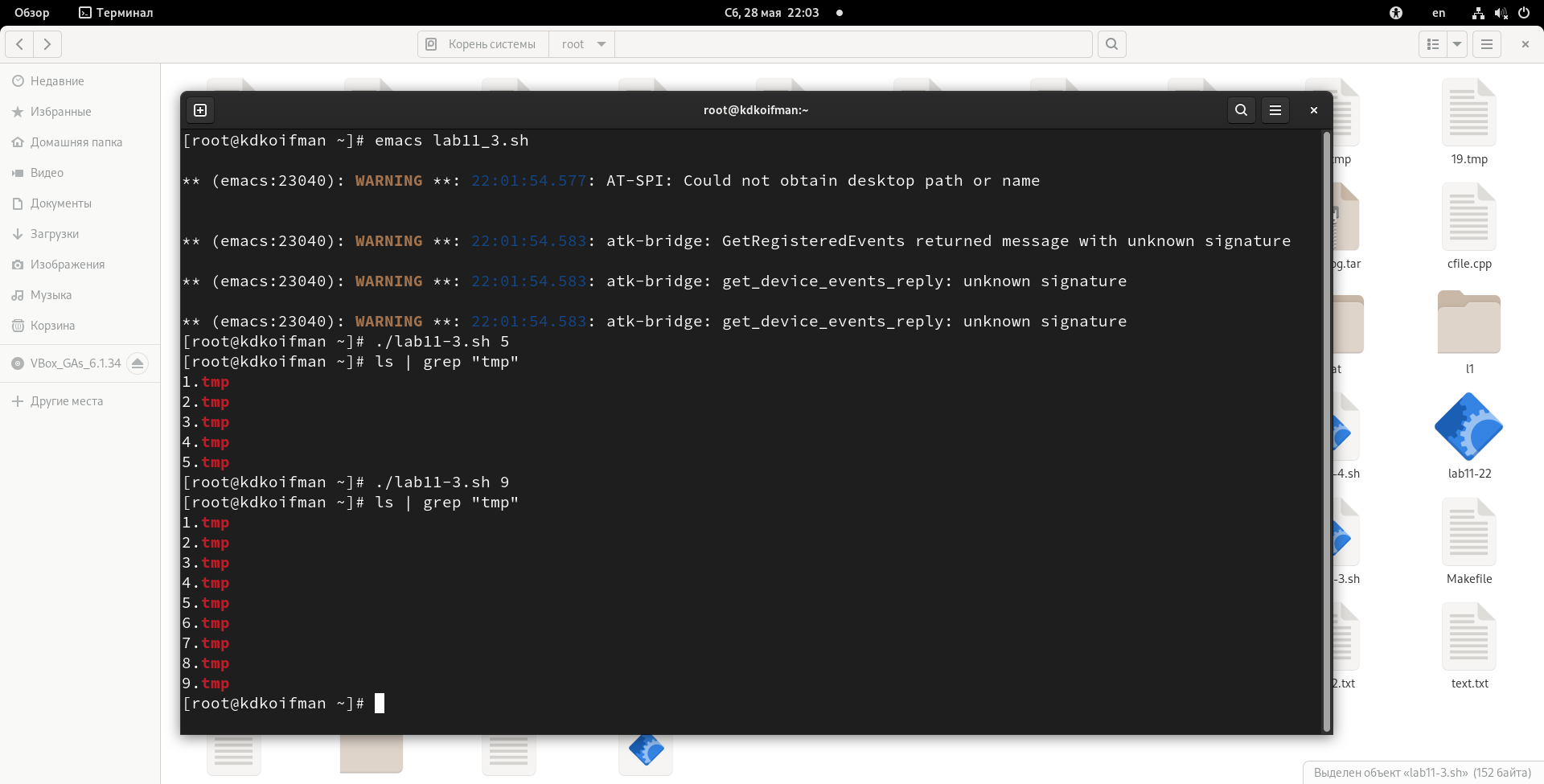
 :–: *рис.6(выполнение командного файла)*

 :–: *рис.7*

## 3 задание.

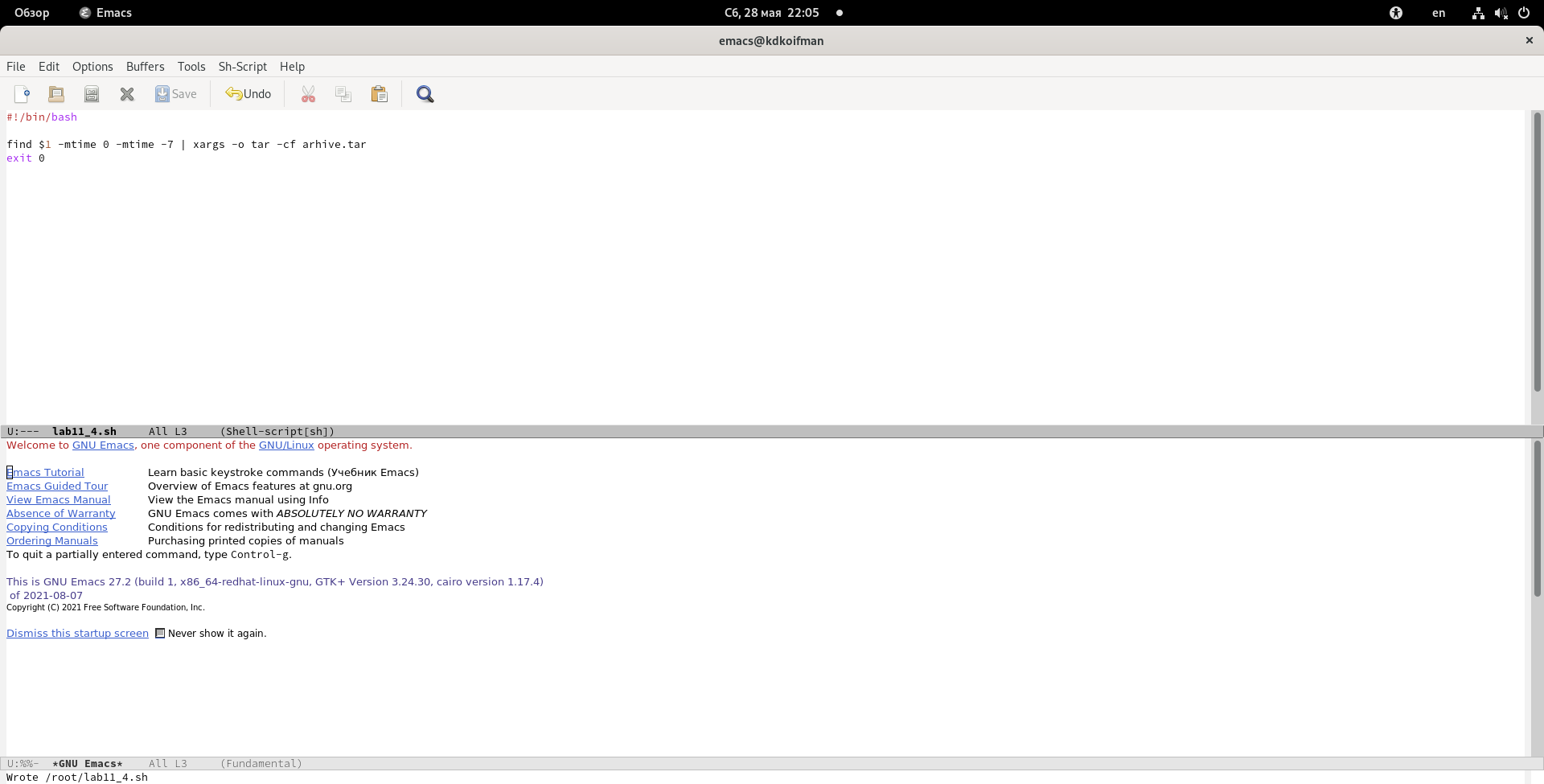
Напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до 𝑁 (например 1.tmp, 2.tmp, 3.tmp,4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют)(рис.8,9)

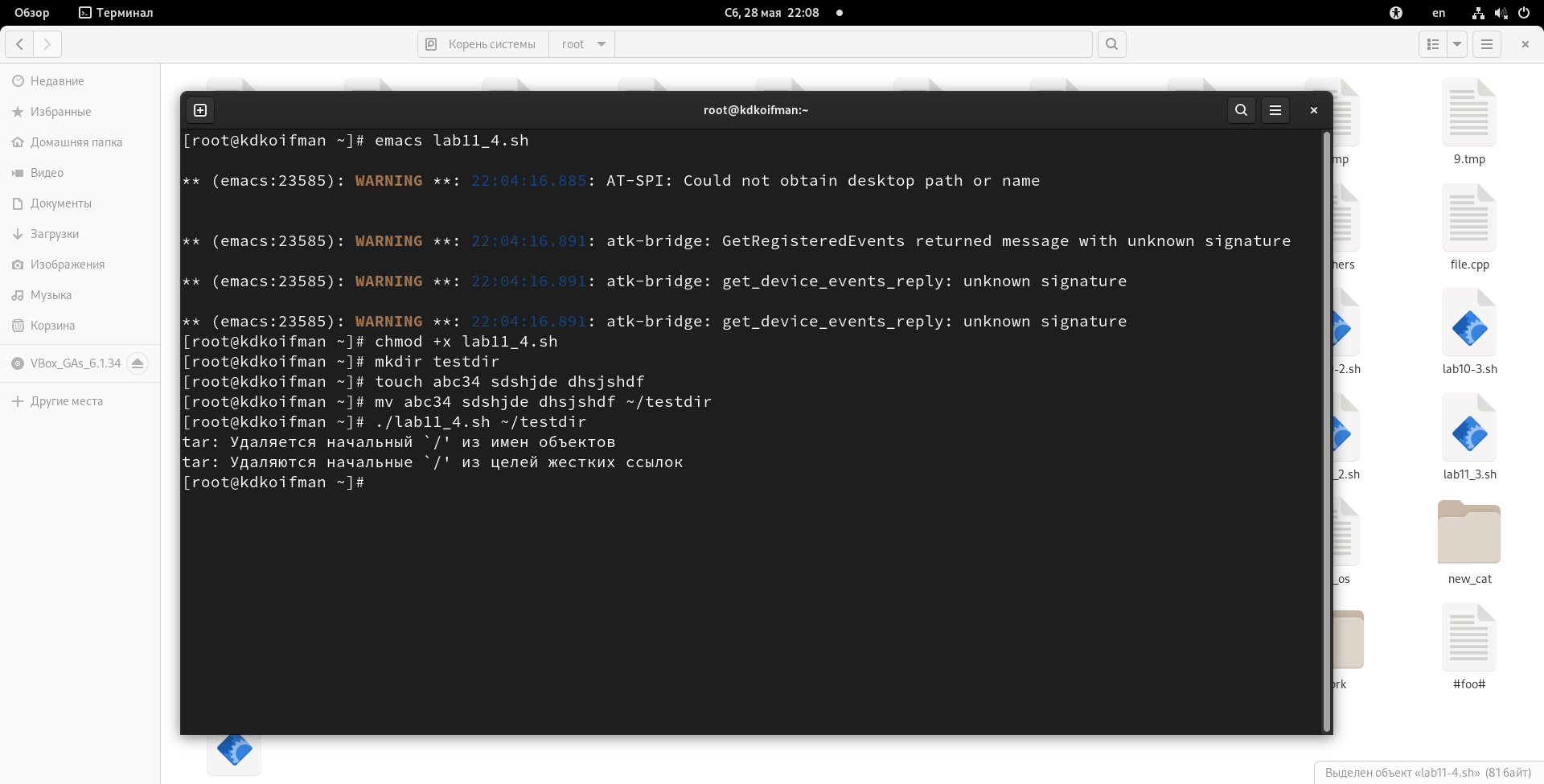
 :–: *рис.8(код командного файла)*

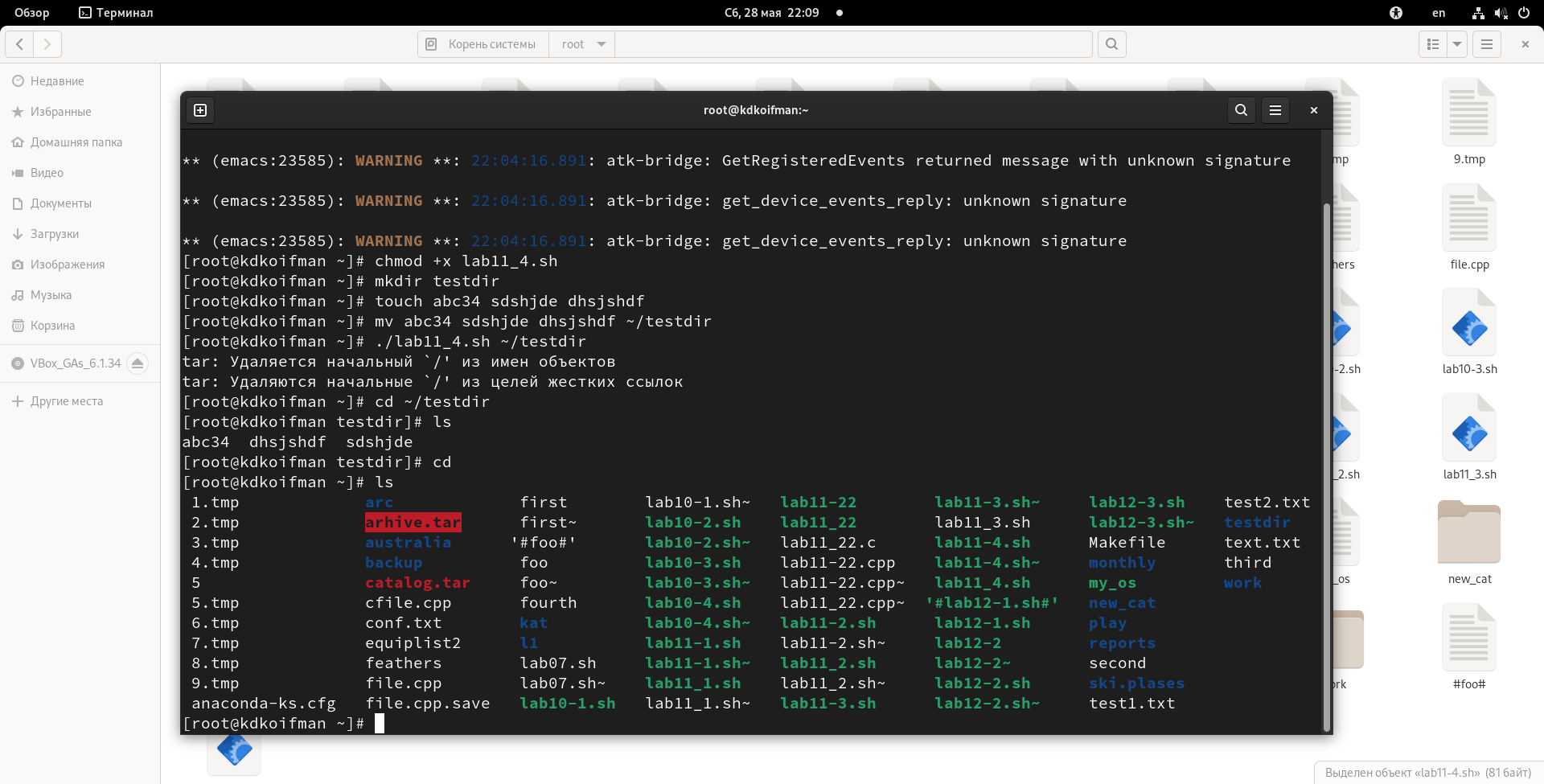
 :–: *рис.9(выполнение командного файла)*

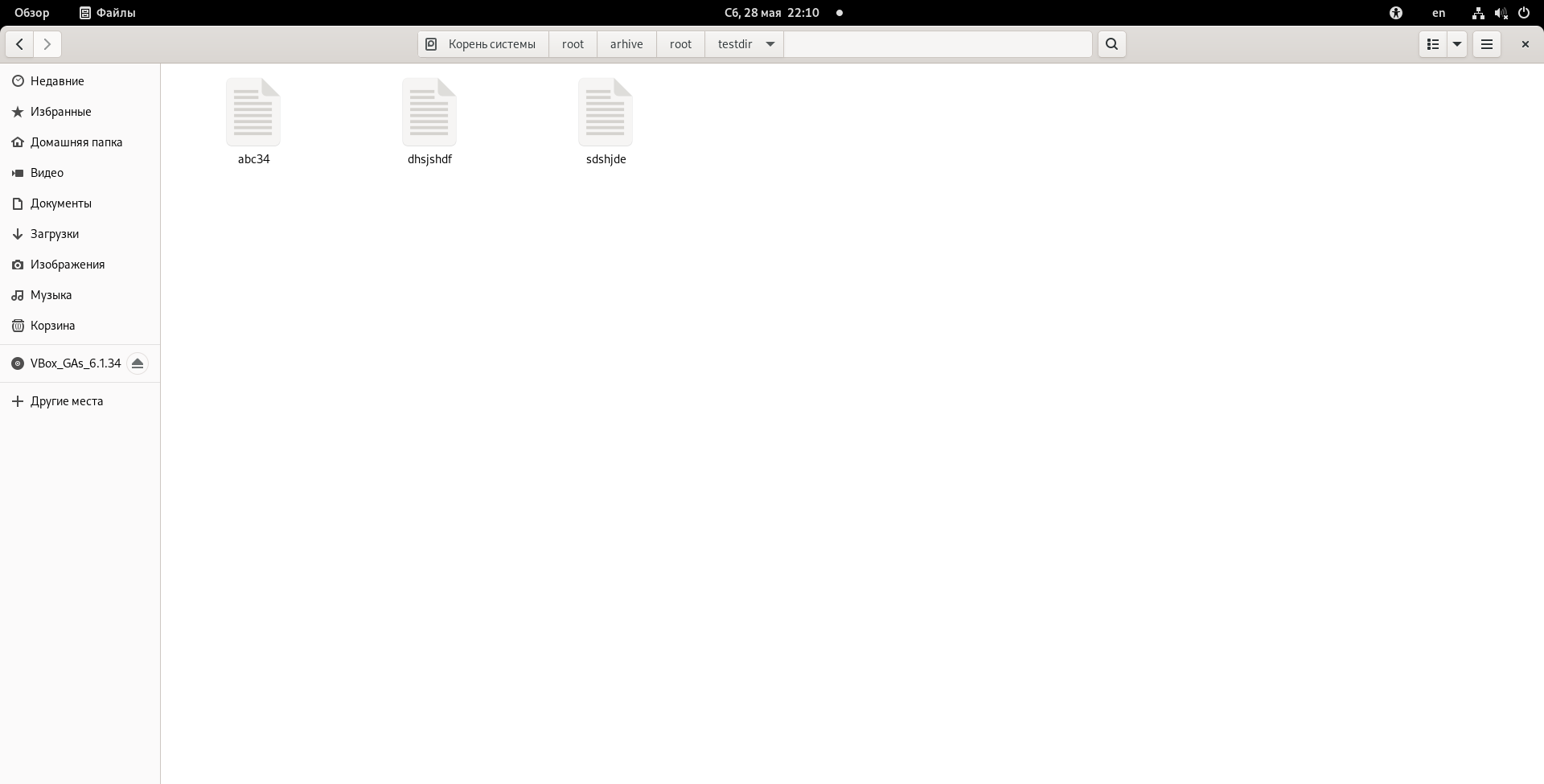
## 4 задание.

Напишем командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицируем его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find)(рис.10,11,12,13)

 :–: *рис.10(код командного файла)*

 :–: *рис.11(выполнение командного файла)*

 :–: *рис.12(архив был создан)*

 :–: *рис.13(содержимое архива)*

## Вывод.

В ходе выполнения лабораторной работы мной были усвоены основные навыки программирования в оболочке ОС UNIX, а также написания комплексных командных файлов с использованием логических управляющих конструкций и циклов.

## Контрольные вопросы.

1. Каково предназначение команды getopts? Команда getopts предназначена для изменения состояния переменных окружения с целью обработки аргументов из командной строки.
2. Какое отношение метасимволы имеют к генерации имён файлов? Метасимволы обрабатываются оболочкой в качестве списка строк, имена которых соответствуют именам файлов, а им в свою очередь соответствует определённая последовательность метасимволов.
3. Какие операторы управления действиями вы знаете? Операторы управления: for, while, if, else, case, switch, then, fi, done, esac.
4. Какие операторы используются для прерывания цикла? Операторы break, return и continue могут прерывать выполнение циклов.
5. Для чего нужны команды false и true? Для работы с условными операторами,циклами и логическими переменными.
6. Что означает строка if test -f mani.$s, встреченная в командном файле?
7. Объясните различия между конструкциями while и until. Оба оператора создают цикл. Однако цикл while будет выполняться до тех пор, пока выражение истинно, а until выполняется пока выражение ложно.