Отчёт по лабораторной работе №10

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Рулев Иван"

Содержание"

# Цель работы

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

# Выполнение лабораторной работы

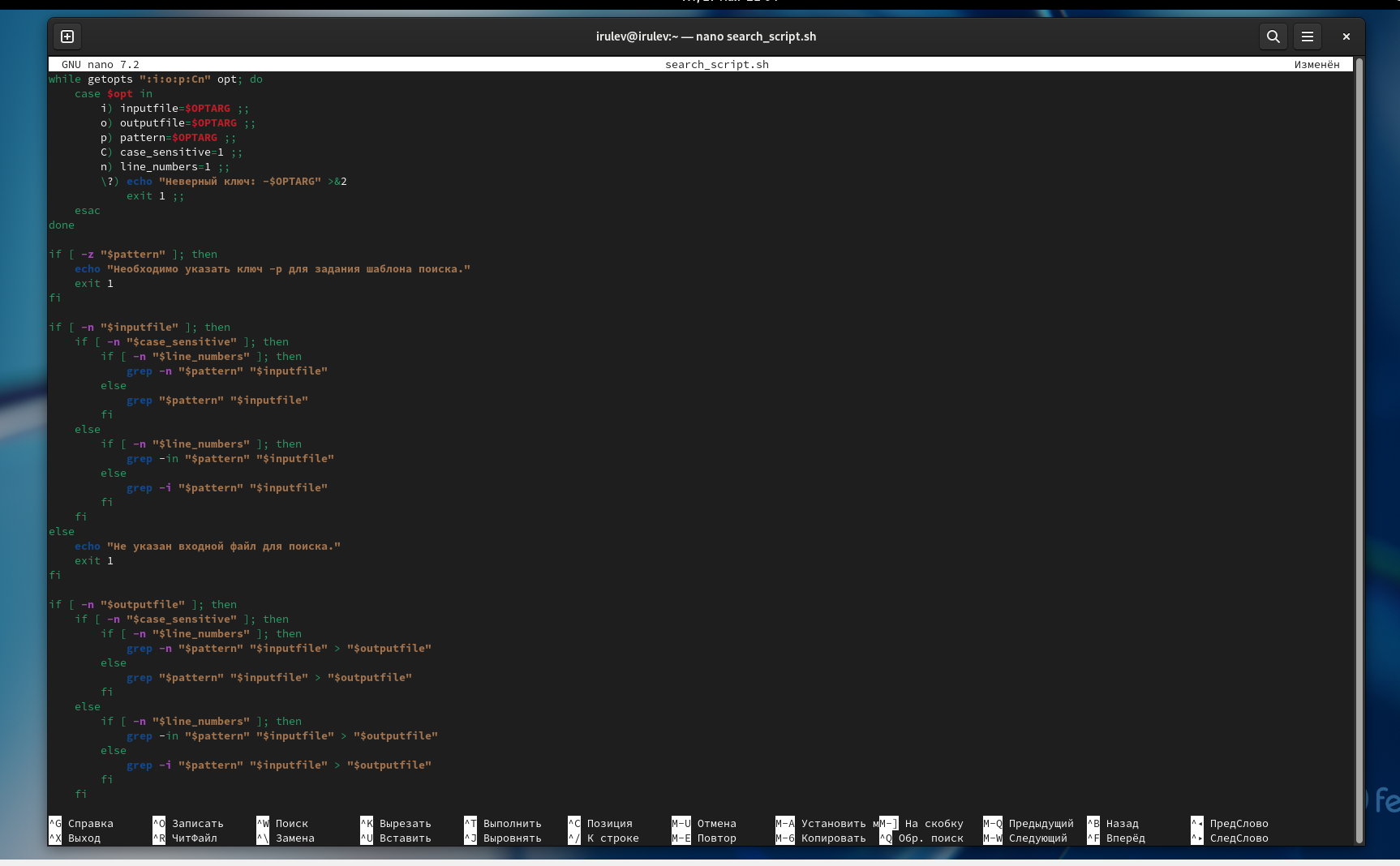
Используя команды getopts grep, написал командный файл, который анализирует командную строку с ключами:

– -iinputfile — прочитать данные из указанного файла;  
– -ooutputfile — вывести данные в указанный файл;

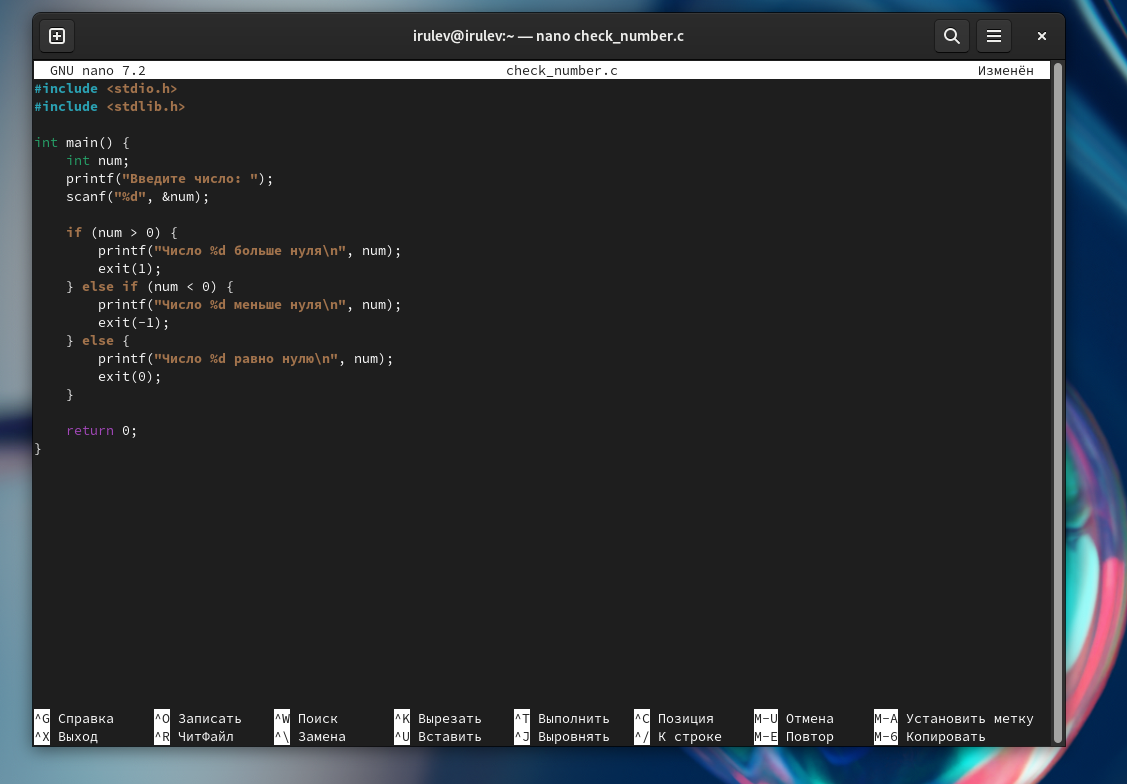
– -pшаблон — указать шаблон для поиска;

– -C — различать большие и малые буквы;   
– -n — выдавать номера строк.

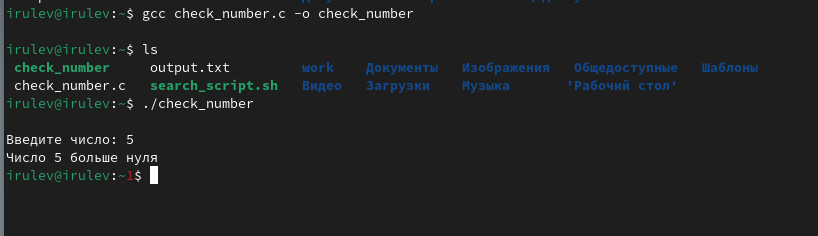
А затем ищет в указанном файле нужные строки, определяемые ключом -p.



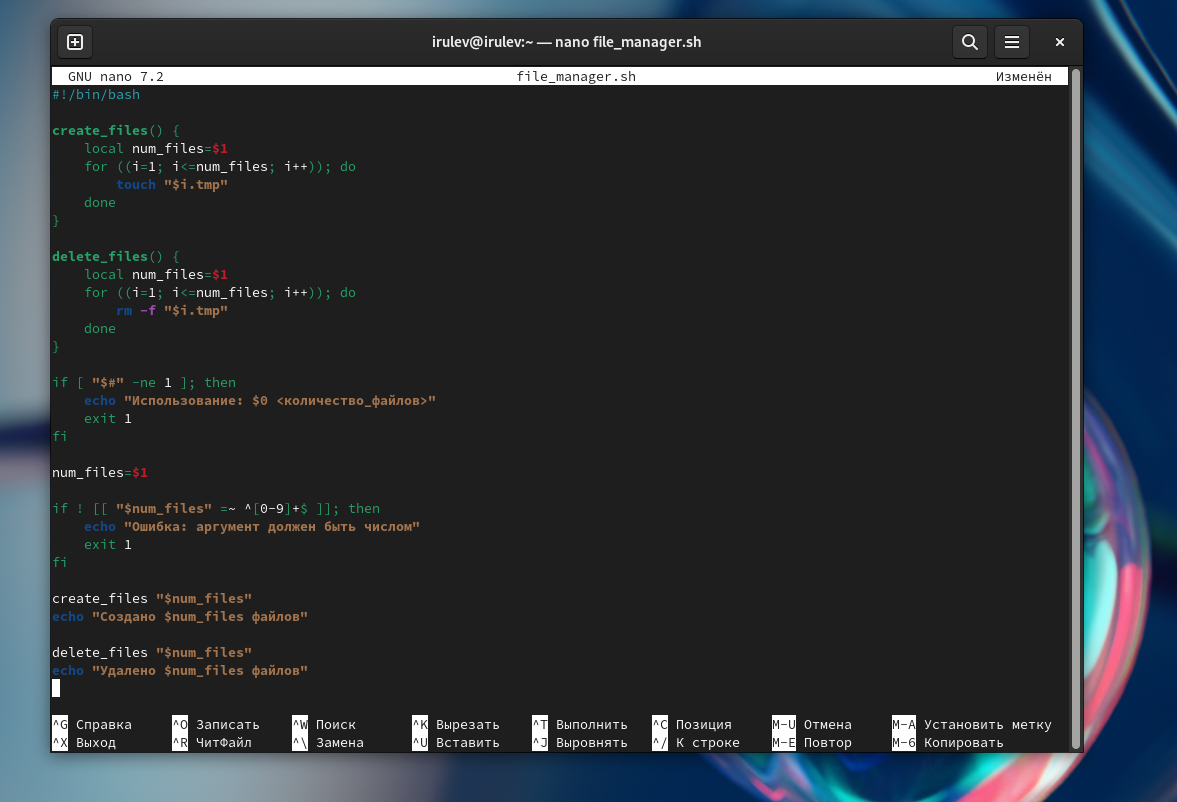
Написал на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции exit(n), передавая информацию в о коде завершения в оболочку. Командный файл вызывает эту программу и, проанализировав с помощью команды $?, выдает сообщение о том, какое число было введено.



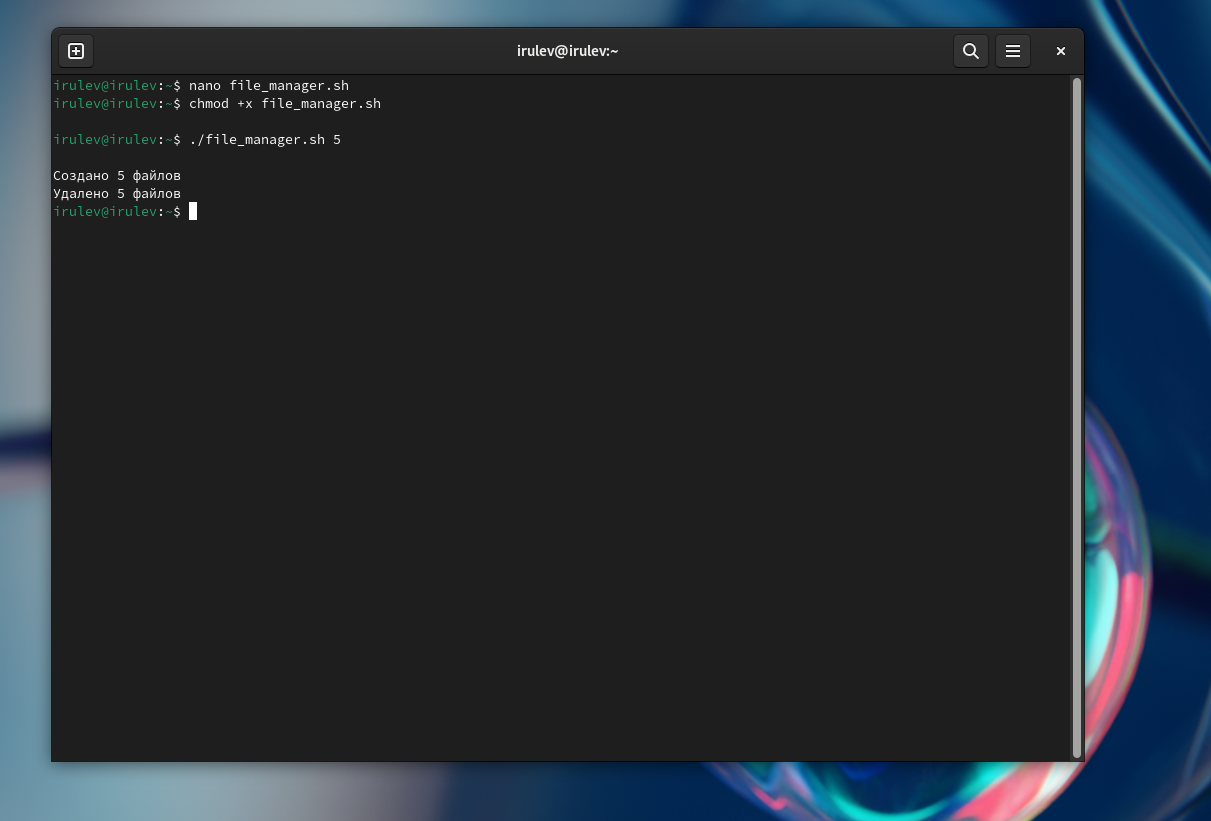
Теперь скомпилируем программу с помощью команды gcc check\_number.c -o check\_number.



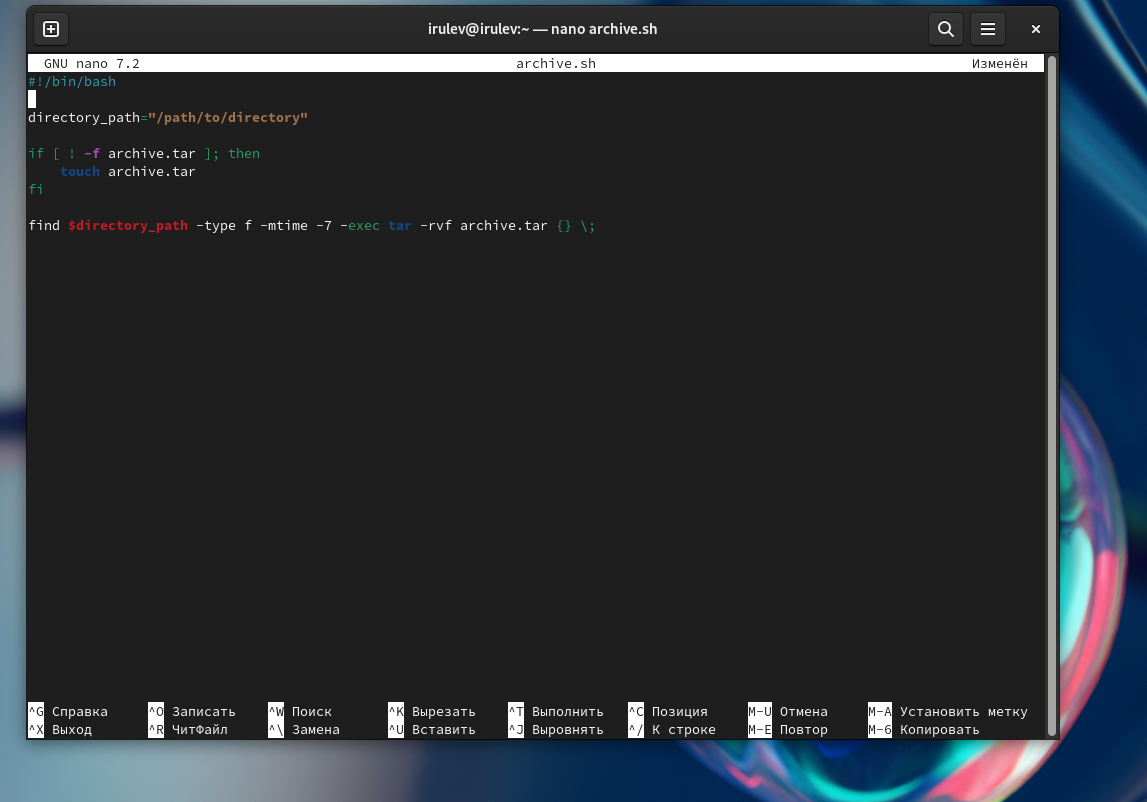
Напишем командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до 𝑁 (например 1.tmp, 2.tmp, 3.tmp,4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если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



Запустим



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).



# Выводы

Изучили основы программирования в оболочке OC UNIX. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# Ответы на контрольные вопросы

1. Команда getopts предназначена для парсинга параметров командной строки в shell-скриптах. Она позволяет извлекать и обрабатывать опции, передаваемые в скрипт, и их аргументы.
2. Метасимволы (\*, ?, [, ], (, ), {, }, ~, .) используются в Linux для генерации имён файлов, известной как globbing. Они позволяют указать шаблоны для поиска файлов, что может быть полезно при выполнении различных операций с файлами, таких как поиск, удаление, копирование и т.д.
3. Операторы управления действиями в Linux включают в себя:
   * && (логическое И) - выполняет вторую команду только если первая команда выполнена успешно.
   * || (логическое ИЛИ) - выполняет вторую команду только если первая команда не выполнена успешно.
   * ; (точка с запятой) - разделяет команды, которые выполняются последовательно.
   * & (амперсанд) - запускает команду в фоне.
   * | (вертикальная черта) - перенаправляет вывод одной команды на вход другой.
4. Операторы, используемые для прерывания цикла, включают в себя:
   * break - прерывает выполнение цикла и продолжает выполнение скрипта после цикла.
   * continue - прерывает текущую итерацию цикла и продолжает выполнение с следующей итерации.
5. Команды false и true используются для возвращения определенного статуса выполнения. true всегда возвращает 0 (успешное выполнение), а false всегда возвращает 1 (неуспешное выполнение). Они могут быть полезны в скриптах для управления потоком выполнения в зависимости от результатов предыдущих операций.
6. Строка if test -f man$s/$i.$s встреченная в командном файле, проверяет, существует ли файл с именем $i.$s в директории man$s. Если файл существует, то условие if будет истинным, иначе - ложным.
7. Конструкции while и until используются для создания циклов в shell-скриптах.
   * while - цикл будет продолжаться, пока условие является истинным.
   * until - цикл будет продолжаться, пока условие является ложным.