Отчёт по лабораторной работе №12

Дисциплина: Операционные системы

Ездаков Егор Андреевич

Содержание

# Цель работы

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

# Задание

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).

# Выполнение лабораторной работы

1. 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.  
  Для данной задачи я создал файл script12\_1.sh и написал соответствующие скрипты (рис. -fig. 1).

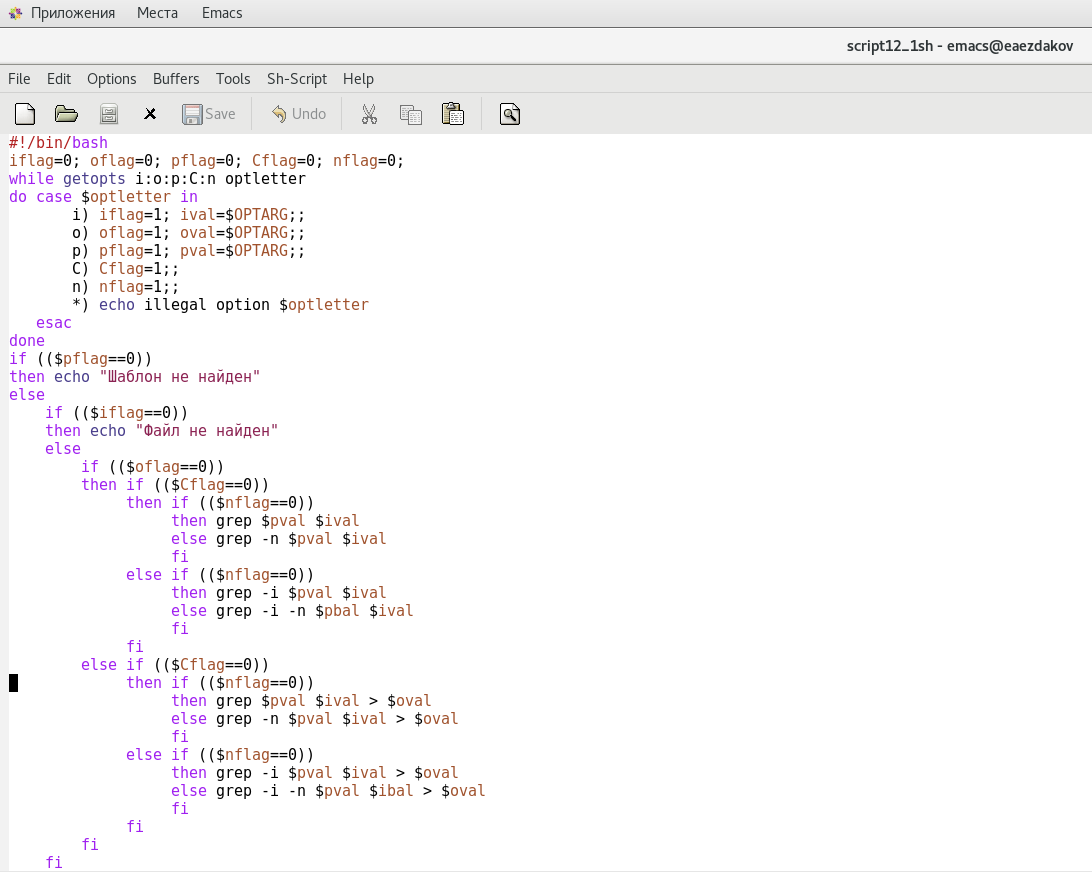


Figure 1: Написал первый скрипт

Далее я проверил работу написанного скрипта, используя различные опции, предварительно добавив право на исполнение файла (команда «chmod +x \*.sh») и создав 2 файла, которые необходимы для выполнения программы: a.txt и b.txt (рис. -fig. 2) (рис. -fig. 3). Скрипт работает корректно.

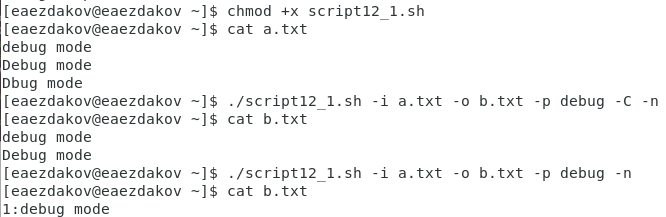


Figure 2: Проверил первый скрипт

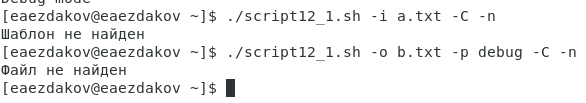


Figure 3: Проверил первый скрипт

1. 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 должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено. Для данной задачи я создал 2 файла: script12\_2.c и script12\_2.sh (рис. -fig. 4) и написал соответствующие скрипты (рис. -fig. 5) (рис. -fig. 6).

Figure 4: Создал файлы

Figure 4: Создал файлы

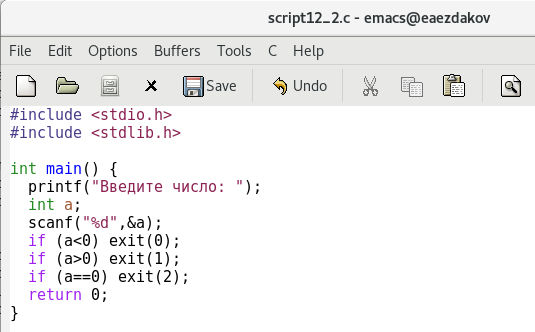


Figure 5: Написал первый скрипт

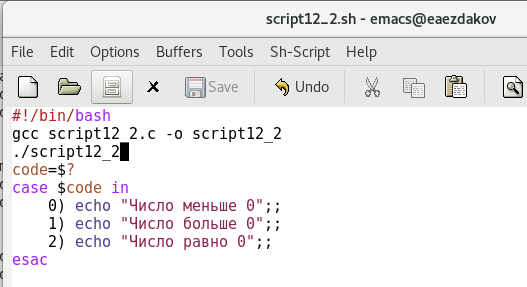


Figure 6: Написал второй скрипт

Далее я проверил работу написанных скриптов (команда «./script12\_2.sh»), предварительно добавив право на исполнение файла (команда «chmod +x \*.sh») (рис. -fig. 7). Скрипты работают корректно.

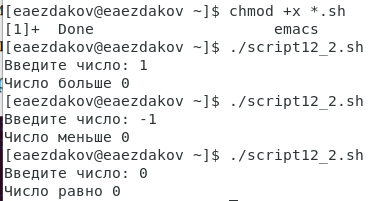


Figure 7: Проверил второй скрипт

1. 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 должен уметь удалять все созданные им файлы (если они существуют). Для данной задачи я создал файл: script12\_3.sh и написал соответствующий скрипт (рис. -fig. 8).

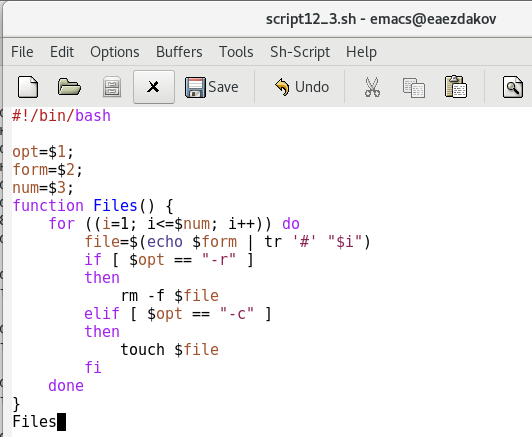


Figure 8: Написал третий скрипт

Далее я проверил работу написанного скрипта (команда «./script12\_3.sh»), предварительно добавив право на исполнение файла (команда «chmod +x \*.sh»). Сначала я создал четыре файла (команда «./script12\_3.sh –c file#.tmp 4»), удовлетворяющие условию задачи, а потом удалил их (команда «./script12\_3.sh –r file#.tmp 4») (рис. -fig. 9). Скрипт работает корректно.

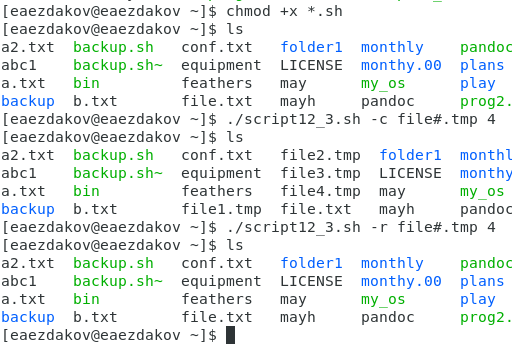


Figure 9: Проверил третий скрипт

1. Написал командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировала его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find). Для данной задачи я создал файл: script12\_4.sh и написал соответствующий скрипт (рис. -fig. 10).

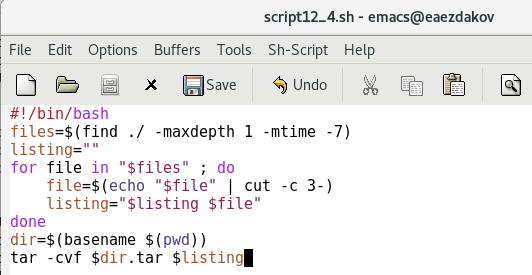


Figure 10: Написал четвёртый скрипт

Далее я проверил работу написанного скрипта (команды «sudo ~/work/2020-2021/os-intro/laboratory/lab12/prog4.sh» и «tar -tf lab12.tar»), предварительно добавив право на исполнение файла (команда «chmod +x \*.sh») (рис. -fig. 11). Скрипт работает корректно.

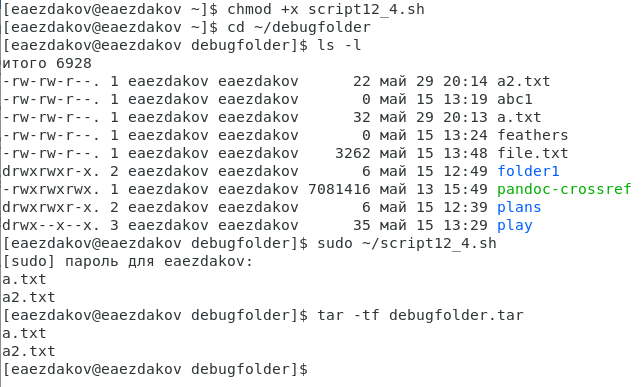


Figure 11: Проверил четвёртый скрипт

# Выводы

В ходе выполнения данной лабораторной работы я изучил основы программирования в оболочке ОС UNIX и научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.