Лабораторная работа №12

Дисциплина: Операционные системы

Галиев Казиз Жарылкасымович

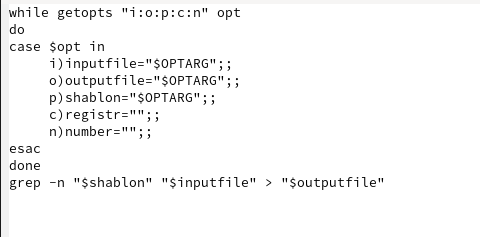
Содержание

# Цель работы

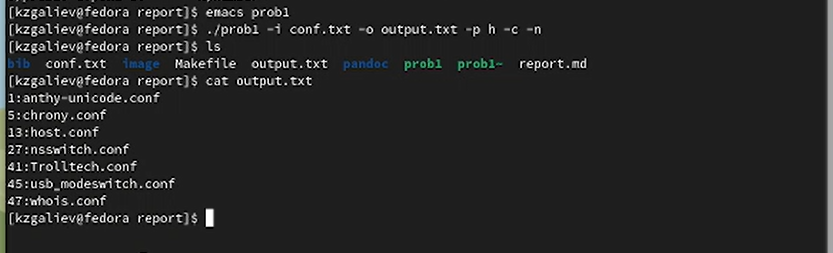
Изучить основы программирования в оболочке ОС UNIX. Научится писать бо- лее сложные командные файлы с использованием логических управляющих кон- струкций и циклов.

# Выполнение лабораторной работы

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 (рис. @fig:001) , (рис. @fig:002) .

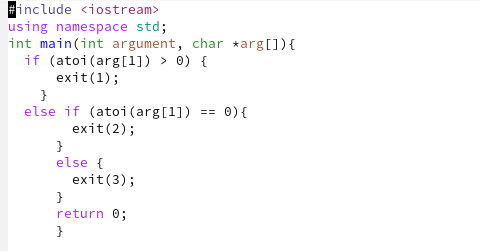


Первый командный файл



Результат выполнения первого командного файла

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 (рис. @fig:003) .



Программа на языке Си

Командный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено (рис. @fig:004) , (рис. @fig:005) .

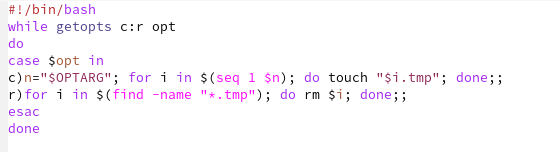


Текст второго командного файла

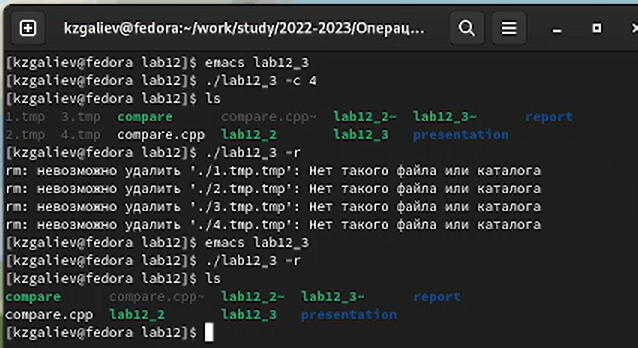


Результат выполнения второго командного файла

1. 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) (рис. @fig:006) , (рис. @fig:007) .



Текст третьего командного файла

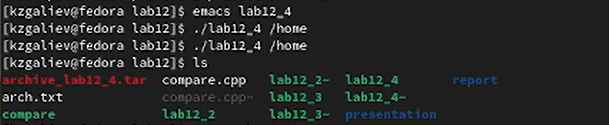


Результат выполнения третьего командного файла

1. 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), (рис. @fig:008), (рис. @fig:009) .



Текст четвертого командного файла

 .

# Выводы

В результате проделанной лабораторной работы я изучил очновы программирования в оболочке ОС Unix и научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# Контрольные вопросы

1. Каково предназначение команды getopts?

Осуществляет синтаксический анализ командной строки, выделяя флаги, и используется для объявления переменных.

1. Какое отношение метасимволы имеют к генерации имён файлов?

Такие символы, как ’ < > \* ? |  ” &, являются метасимволами и имеют для командного процессора специальный смысл. Снятие специального смысла с метасимвола называется экранированием метасимвола. Экранирование может быть осуществлено с помощью предшествующего метасимволу символа , который, в свою очередь, является метасимволом. Для экранирования группы метасимволов нужно заключить её в одинарные кавычки. Строка, заключённая в двойные кавычки, экранирует все метасимволы, кроме $, ’ , , “.

1. Какие операторы управления действиями вы знаете?

Язык программирования bash предоставляет возможность использовать такие управляющие конструкции, как for, case, if и while, until. С точки зрения командного процессора эти управляющие конструкции являются обычными командами и могут использоваться как при создании командных файлов, так и при работе в интерактивном режиме. Команды, реализующие подобные конструкции, по сути, являются операторами языка программирования bash. Поэтому при описании языка программирования bash термин оператор будет использоваться наравне с термином команда.

1. Какие операторы используются для прерывания цикла?

Команда break завершает выполнение цикла, а команда continue завершает данную итерацию блока операторов.

1. Для чего нужны команды false и true?

Команда true, которая всегда возвращает код завершения, равный нулю (т.е. истина), и команда false, которая всегда возвращает код завершения, не равный нулю (т. е. ложь), используются только совместно с управляющими конструкциями языка программирования bash

1. Объясните различия между конструкциями while и until. При замене в операторе цикла while служебного слова while на until условие, при выполнении которого осуществляется выход из цикла, меняется на противоположное. В остальном оператор цикла while и оператор цикла until идентичны. В обобщённой форме оператор цикла until выглядит следующим образом: until список-команд do список-команд done.

# Список литературы