Отчёт по лабораторной работе №13

Операционные системы

Чистов Даниил Максимович

Содержание

# 1 Цель работы

Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

# 2 Задания

1. Программа 1
2. Программа 2
3. Программа 3
4. Программа 4

# 3 Выполнение лабораторной работы

# 4 Программа 1

Задание: “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.”

Приступаю к выполнению работы. Создаю файл с кодом и пишу код (сама программа будет показана позже), выполняю - готово (рис. 1).

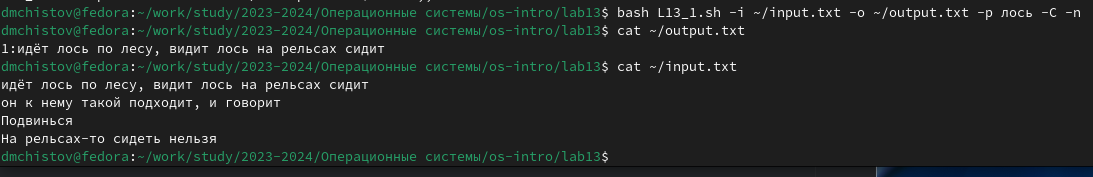


Рис. 1: Задание 1 - выполнено

Код программы 1: утилитой getopts считываем коды, а дальше утититой case рассматриваем каждый случай. Затем утилитой grep считываем и используем output.txt (рис. 2).

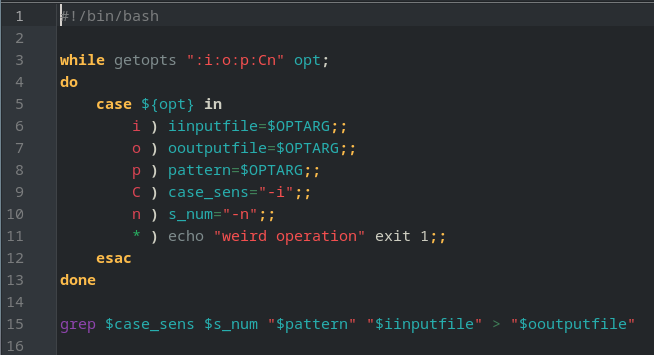


Рис. 2: Задание 1 - программа

# 5 Программа 2

Задание: “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. Команд ный файл должен вызывать эту программу и, проанализировав с помощью команды $?, выдать сообщение о том, какое число было введено”

Приступаю к выполнению - создаю, даю права, пишу код, исправляю ошибки, запускаю - работает (рис. 3).

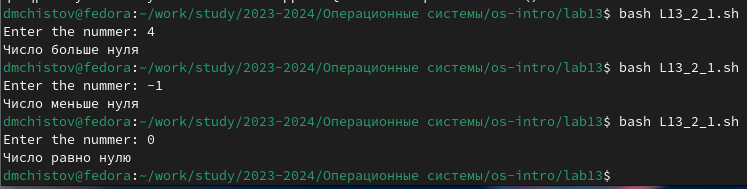


Рис. 3: Задание 2 - выполнено

Код программы 2\_1 (bash): открываем файл с кодом на C, запускаем его и получаем код, утилитой case в зависимости от кода возвращаем нужное сообщение (рис. 4).

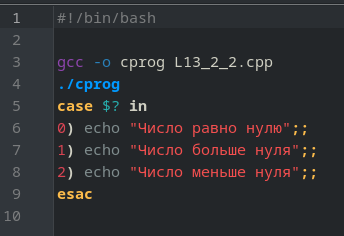


Рис. 4: Задание 2 - программа 1

Программа 2\_2: Если число больше нуля возвращаем 1, если меньше - 2, если равно - 0 (рис. 5).

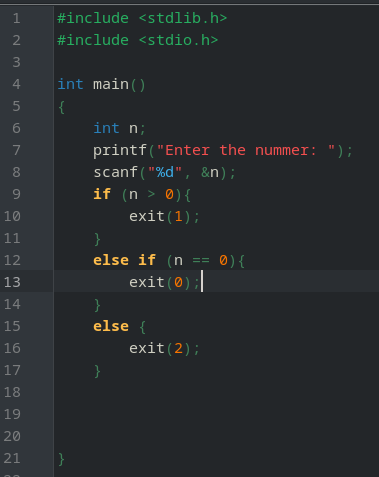


Рис. 5: Задание 2 - программа 2

# 6 Программа 3

Задание: “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 ко мандный файл должен уметь удалять все созданные им файлы (если они существуют).”

Создаю файл, даю права, начинаю писать код, всё работает (рис. 6).

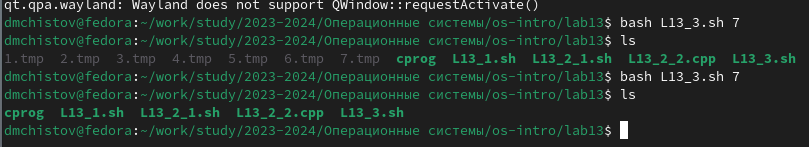


Рис. 6: Задание 3 - выполнено

Код программы 3: Иду по циклу от 1 до заданного числа, проверяю утилитой test есть ли такие файлы под названием “Номер”.tmp. Если есть удаляю, если нет, создаю (рис. 7).



Рис. 7: Задание 3 - программа

# 7 Программа 4

Задание: “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).”

Создаю файл, даю права, начинаю писать код. Готово (рис. 8).

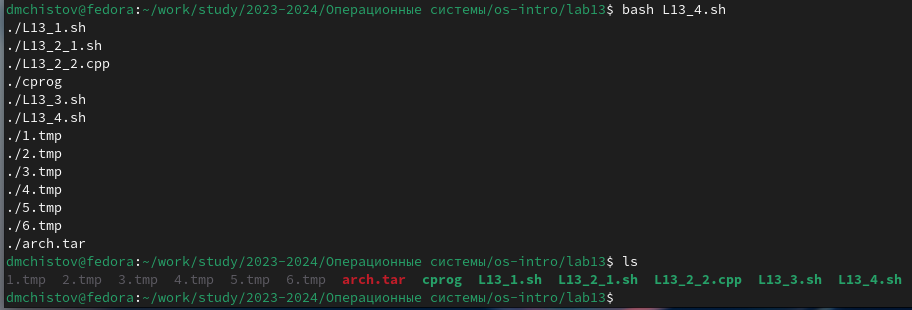


Рис. 8: Задание 4 - готово

Код программы 4: Использую конвеер. В первой части ищу утилитой find все файлы, которые были модифцированы менее 7 дней назад (-mtime -7), вывожу их. Во второй части принимаю этот вывод, архивирую (рис. 9).

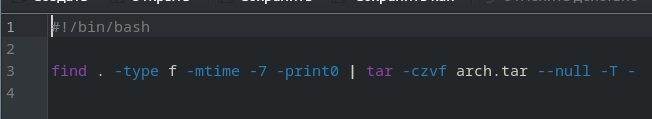


Рис. 9: Задание 4 - программа

# 8 Выводы

В результате выполненения данной работы я изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.

# Список литературы

[Лабораторная работы №13](https://esystem.rudn.ru/pluginfile.php/2288099/mod_resource/content/5/011-lab_shell_prog_2.pdf) [Команда find](https://habr.com/ru/companies/alexhost/articles/525394/)