Лабораторная работа-11

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Доленко Дарья Васильевная НБИбд-01-21

Содержание

# 1 Цель работы

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

# 2 Выполнение лабораторной работы

Используя команды getopts grep, написала командный файл, который анализирует командную строку с ключами: -iinputfile; -ooutputfile; -pшаблон; -C; -n, затем ищет в указанном файле нужные строки,определяемые ключом -p. (рис. 1 2 3)

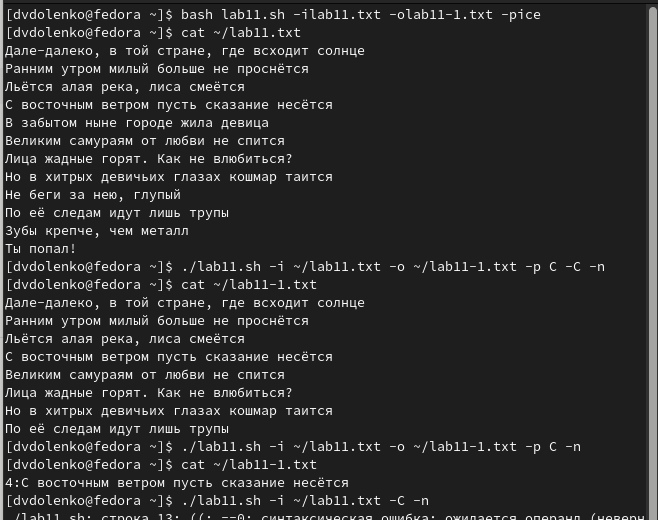


Рис. 1: Пример работы программы

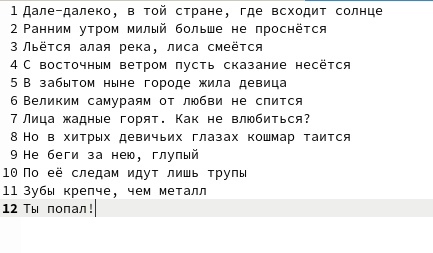


Рис. 2: Содержание текстового файла

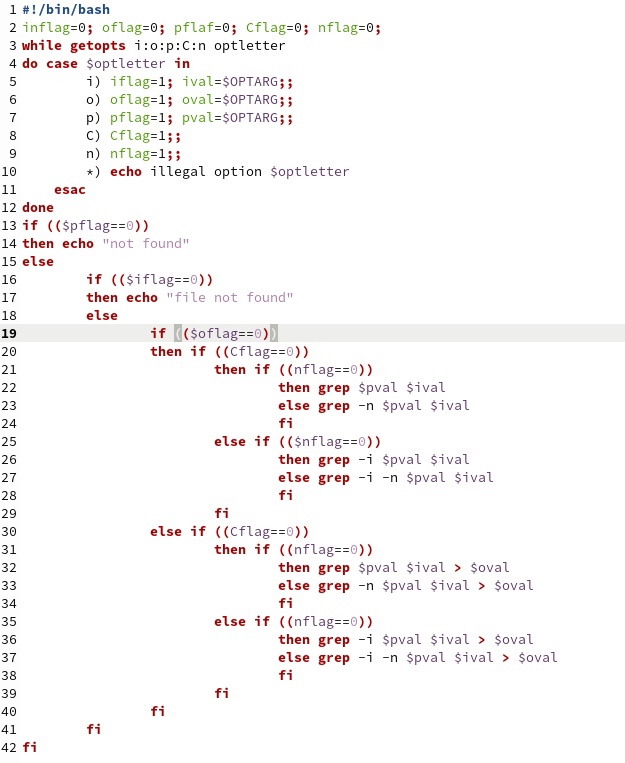


Рис. 3: Текст программы

Написала на языке Си программу, которая вводитчисло и определяет, являетсяли 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 отом, какое число было введено. (рис. 4 5 6)

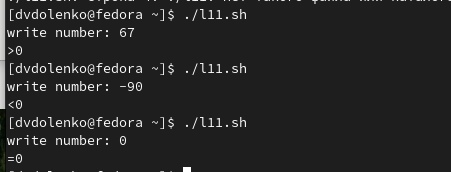


Рис. 4: Пример работы программы

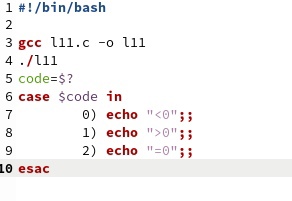


Рис. 5: Файл sh

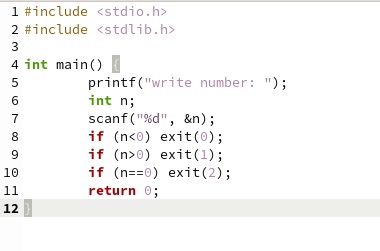


Рис. 6: Файл С

Написала 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. 7 8 9)

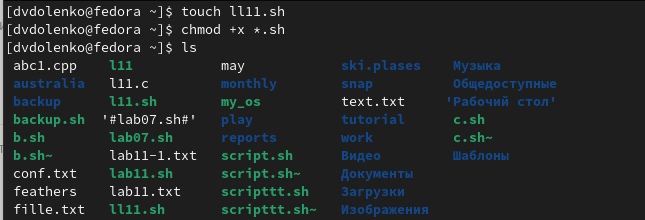


Рис. 7: Пример работы программы (состояние до применения)

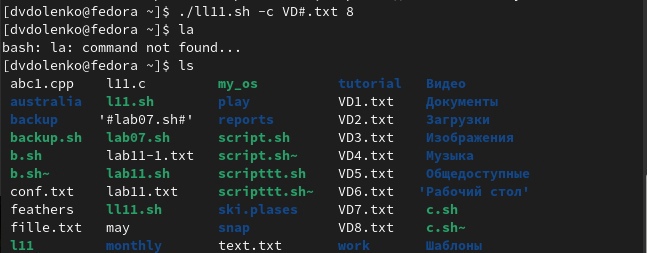


Рис. 8: Пример работы программы (после применения)

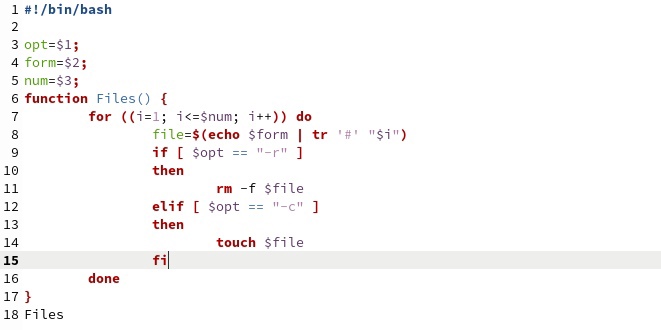


Рис. 9: Текст программы

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

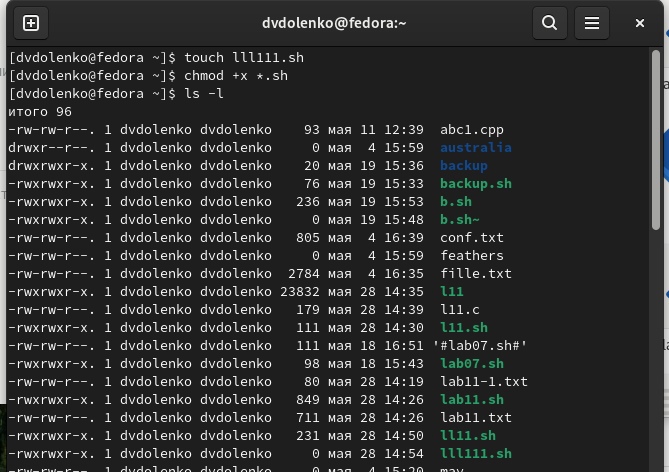


Рис. 10: До применения программы

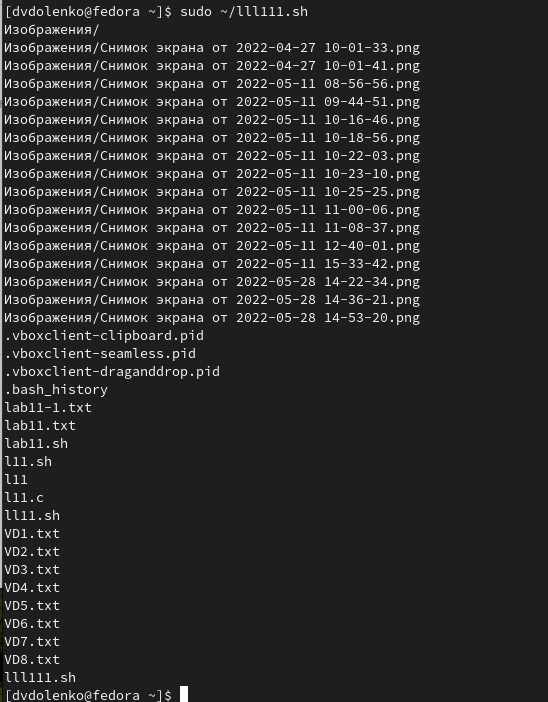


Рис. 11: После применения программы



Рис. 12: Текст программы

Рис. 13: Команды архивации

Рис. 13: Команды архивации

# 3 Вывод

В ходе данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX, научилась писать более сложные командные файлы с использованиемлогических управляющих конструкций и циклов.

# 4 Ответы на контрольные вопросы:

1. Команда getopts является встроенной командой командной оболочки bash, предназначенной для разбора параметров сценариев. Она обрабатывает исключительно однобуквенные параметры как с аргументами, так и без них и этого вполне достаточно для передачи сценариям любых входных данных.
2. При генерации имен используют метасимволы:

* произвольная (возможно пустая) последовательность символов; ? один произвольный символ; […] любой из символов, указанных в скобках перечислением и/или с указанием диапазона; cat f\* выдаст все файлы каталога, начинающиеся с “f”; cat f выдаст все файлы, содержащие “f”; cat program.? выдаст файлы данного каталога с однобуквенными расширениями, скажем “program.c” и “program.o”, но не выдаст “program.com”; cat [a-d]\* выдаст файлы, которые начинаются с “a”, “b”, “c”, “d”. Аналогичный эффект дадут и команды “cat [abcd]” и “cat [bdac]”.

1. Операторы && и || являются управляющими операторами. Если в командной строке стоит command1 && command2, то command2 выполняется в том, и только в том случае, если статус выхода из команды command1 равен нулю, что говорит об успешном ее завершении. Аналогично, если командная строка имеет вид command1 || command2, то команда command2 выполняется тогда, и только тогда, когда статус выхода из команды command1 отличен от нуля.
2. Оператор break завершает выполнение ближайшего включающего цикла или условного оператора, в котором он отображается.
3. Команда true всегда возвращает ноль в качестве выходного статуса для индикации успеха. Команда false всегда возвращает не-ноль в качестве выходного статуса для индикации неудачи. Во всех управляющих конструкциях в качестве логического значения используется код возврата из программы, указанной в качестве условия. Код возврата 0 – истина, любое другое значение – ложь. Программа true – всегда завершается с кодом 0, false – всегда завершается с кодом 1.
4. Введенная строка означает условие существования файла mani.$s
5. Цикл While выполняется до тех пор, пока указанное в нем условие истинно. Когда указанное условие становится ложным - цикл завершается. Цикл Until выполняется до тех пор, пока указанное в нем условие ложно.