Лабораторная работа № 12

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Маметкадыров Ынтымак

# 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

1. Написали на языке Си программу (рис. 1), 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 (рис. 2) вызывает эту программу и, проанализировав с помощью команды $?, выдает сообщение о том, какое число было введено (рис. 3).

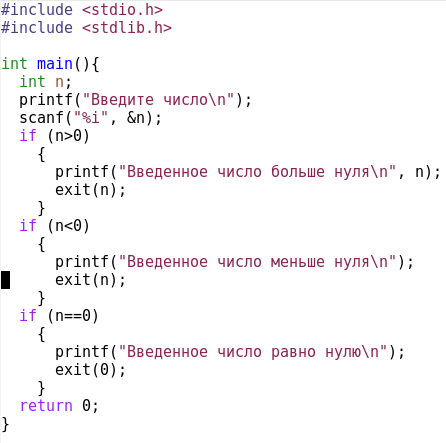


Рис. 1. Листинг программы compare.c

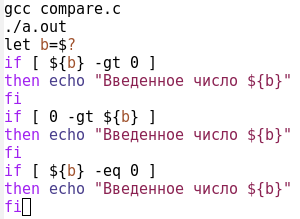


Рис. 2. Листинг командного файла compare.sh

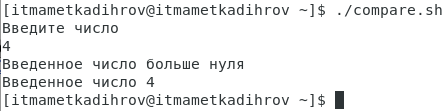


Рис. 3. Работа командного файла compare.sh

1. Написали командный файл files.sh (рис. 4), создающий указанное число файлов, пронумерованных последовательно от 1 до N. Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Командный файл также умеет удалять все созданные им файлы, для этого нужно при запросе об удалении нажать 1, иначе 0. Передали в качестве параметра командному файлу число 5, в результате создались файлы 1.tmp, 2.tmp, 3.tmp, 4.tmp, 5.tmp (рис. 5). Попробовали удалить все созданные файлы (рис. 6).

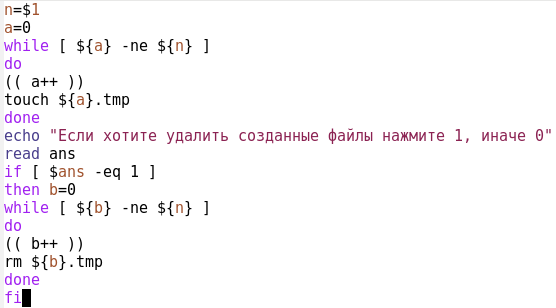


Рис. 4. Командный файл files.sh

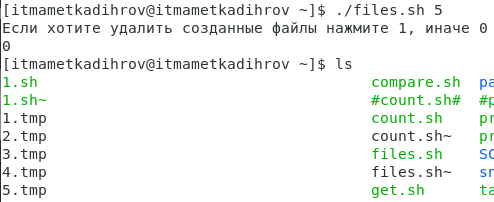


Рис. 5. Создание файлов

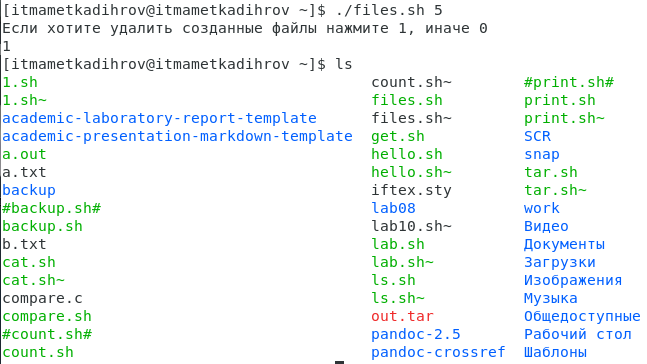


Рис. 6. Удаление файлов

1. Написали tar -cvf out.tar work в командный файл tar.sh, который с помощью команды tar запаковывает в архив все файлы в указанной директории work (рис. 7).

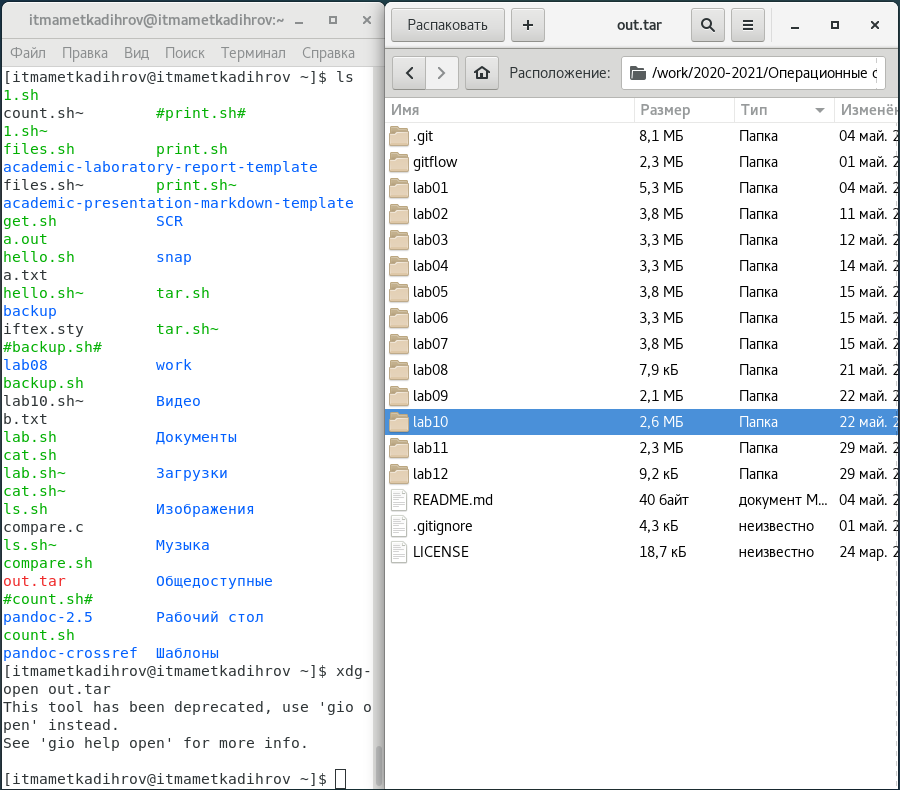


Рис. 7. Результат работы командного файла tar.sh

1. Модифицировали его так (рис. 8), чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (рис. 9).

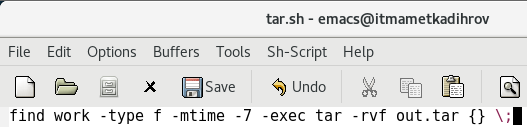


Рис. 8. Модифицированный командный файл tar.sh

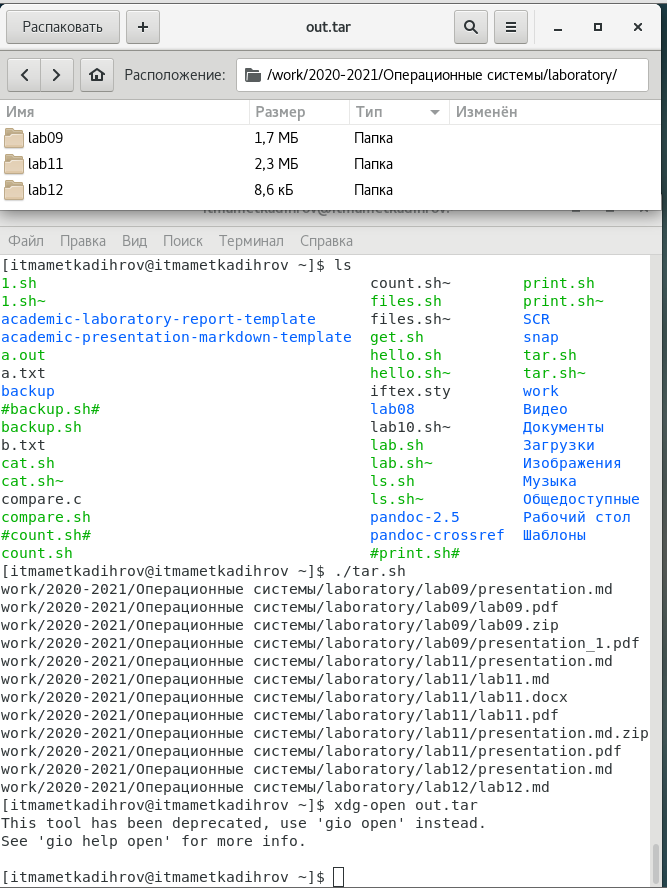


Рис. 9. Результат работы командного файла tar.sh

# Вывод

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

# Ответы на контрольные вопросы

1. getopts - осуществляет синтаксический анализ командной строки, выделяя флаги, и используется для объявления переменных.
2. Такие символы, как ’ < > \* ? |  " &, являются метасимволами и имеют для командного процессора специальный смысл.
3. Управляющие операторы, как for, case, if и while.
4. Команды прерывания break и continue.
5. Используются для возвращения кода завершения. true всегда возвращает код завершения, равный нулю (т.е. истина), а команда false всегда возвращает код завершения, не равный нулю (т. е. ложь).
6. Возвращается истину, если существует файл.
7. 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.