



Отчет по лабораторной работе №20 по курсу 1
Студент группы M80-101Б-22, № по списку 1
Контакты e-mail timur.buchkin@mail.ru
Работа выполнена: «14» февраля 2023 г.
Преподаватель: каф. 806 Крылов С. С.
Входной контроль знаний с оценкой ---
Отчет сдан «21» февраля 2023 г., итоговая оценка ---
Подпись преподавателя ---

1. Тема: Стандартные утилиты UNIX для обработки файлов
2. Цель работы: Получить практические навыки работы с утилитами UNIX
3. Задание: Запротоколировать результат работы основных утилит UNIX для работы с файлами
4. Оборудование ПЭВМ студента, если использовалось: 2,5 GHz 8-ядерный процессор Intel Core i5. Монитор: Универсальный монитор PnP
5. Программное обеспечение ЭВМ студента, если использовалось: Операционная система семейства: Windows, наименование: Windows 11, интерпретатор команд: WSL2
Система программирования: нет
Редактор текстов: Sublime text 3
Утилиты операционной системы: wc, dd, diff, grep, join, sort, tail, tee, tr, uniq, od, cut, sum, tar, gzip, bzip2, head, iconv, md5sum, du, file, touch, file, mktemp
6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями):
Алгоритм:
 - (a) Выбрать команду
 - (b) Изучить её
 - (c) Опробовать в терминале
 - (d) Результат запротоколировать
7. Сценарий выполнения работы [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].
 - (a) wc - Подсчитывает количество слов, строк, байтов во входном потоке данных.
Синтаксис: \$ wc [параметры] файл Параметры:
 - -c -bytes Отобразить размер объекта в байтах
 - -m -count Показать количество символов в объекте
 - -l -lines Вывести количество строк в объекте

- -w -words Отобразить количество слов в объекте
- (b) dd - копирует файлы из одного места в другое в виде двоичных данных.
Синтаксис: \$ dd if=источниккопирования of=местоназначения параметры.
Параметры:
- bs - указывает сколько байт читать и записывать за один раз;
 - cbs - сколько байт нужно записывать за один раз;
 - count - скопировать указанное количество блоков, размер одного блока указывается в параметре bs;
 - conv - применить фильтры к потоку данных;
 - ibs - читать указанное количество байт за раз;
 - obs - записывать указанное количество байт за раз;
 - seek - пропустить указанное количество байт в начале устройства для чтения;
 - skip - пропустить указанное количество байт в начале устройства вывода;
 - status - указывает насколько подробным нужно сделать вывод;
 - iflag, oflag - позволяет задать дополнительные флаги работы для устройства ввода и вывода, основные из них: nocache, nofollow
- (c) diff -это утилита командной строки, которая позволяет сравнивать два файла построчно. Он также может сравнивать содержимое каталогов.
Синтаксис: \$ diff опции файл1 файл2 Опции:
- -q - выводить только отличия файлов;
 - -s - выводить только совпадающие части;
 - -c - выводить нужное количество строк после совпадений;
 - -u - выводить только нужное количество строк после отличий;
 - -y - выводить в две колонки;
 - -e - вывод в формате ed скрипта;
 - -n - вывод в формате RCS;
 - -a - сравнивать файлы как текстовые, даже если они не текстовые;
 - -t - заменить табуляции на пробелы в выводе;
 - -l - разделить на страницы и добавить поддержку листания;
 - -r - рекурсивное сравнение папок;
 - -i - игнорировать регистр;
 - -E - игнорировать изменения в табуляциях;
 - -Z - не учитывать пробелы в конце строки;
 - -b - не учитывать пробелы;
 - -B - не учитывать пустые строки.
- (d) grep - эта утилита позволяет пользователям ввести поиск строки в файле.
Синтаксис: \$ grep [опции] шаблон [имя файла...]
Параметры:
- -b - показывать номер блока перед строкой;
 - -c - подсчитать количество вхождений шаблона;
 - -h - не выводить имя файла в результатах поиска внутри файлов Linux;
 - -i - не учитывать регистр;
 - -l - отобразить только имена файлов, в которых найден шаблон;
 - -n - показывать номер строки в файле;
 - -s - не показывать сообщения об ошибках;
 - -v - инвертировать поиск, выдавать все строки кроме тех, что содержат шаблон;
 - -w - искать шаблон как слово, окружённое пробелами;

- -e - использовать регулярные выражения при поиске;
 - -An - показать вхождение и n строк до него;
 - -Bn - показать вхождение и n строк после него;
 - -Cn - показать n строк до и после вхождения;
- (e) join - это утилита, объединяющая строки двух упорядоченных текстовых файлов на основе наличия общего поля.
Синтаксис: \$ join [параметры] файл1 файл2
Параметры:
- -a - печатать несопоставимые строки;
 - -v - выводит только строки, которым не нашлось пары;
 - -1 - указать, какое поле использовать в 1 файле;
 - -2 - указать, какое поле использовать во 2 файле;
- (f) sort - это утилита, выводящая отсортированное слияние указанных файлов на стандартный вывод.
Синтаксис: \$ sort опции файл
Параметры:
- -b - не учитывать пробелы
 - -d - использовать для сортировки только буквы и цифры
 - -i - сортировать только по ASCII символам
 - -n - сортировка строк linux по числовому значению
 - -r - сортировать в обратном порядке
 - -c - проверить был ли отсортирован файл
 - -o - вывести результат в файл
 - -u - игнорировать повторяющиеся строки
 - -m - объединение ранее отсортированных файлов
 - -k - указать поле по которому нужно сортировать строки, если не задано, сортировка выполняется по всей строке.
 - -f - использовать в качестве разделителя полей ваш символ вместо пробела.
- (g) tail - это утилита выводящая последние 10 строк файла.
Синтаксис: \$ tail опции файл
Параметры:
- -с - выводить указанное количество байт с конца файла;
 - -f - обновлять информацию по мере появления новых строк в файле;
 - -n - выводить указанное количество строк из конца файла;
 - -pid - используется с опцией -f, позволяет завершить работу утилиты, когда завершится указанный процесс;
 - -q - не выводить имена файлов;
 - -retry - повторять попытки открыть файл, если он недоступен;
 - -v - выводить подробную информацию о файле;
- (h) tee - эта утилита выводит на экран, или же перенаправляет выходной материал команды и копирует его в файл.
Синтаксис: \$ tee опции файл
Параметры:
- -a или -append - Используется для записи вывода в конец существующего файла.
 - -i или -ignore-interrupts - Используется, чтобы игнорировать прерывающие сигналы.
 - -help - Используется для показа всех возможных операций.
 - -version - Используется для показа текущей версии этой команды.

- (i) `tr` - это утилита, которая переводит, удаляет и сжимает символы из стандартного ввода и записывает результат в стандартный вывод.

Синтаксис: `$ tr [КЛЮЧ]... НАБОР1 [НАБОР2]`

Параметры:

- `-c, -C, -complement` Сначала получить дополнение НАБОРА1
- `-d, -delete` Удалить знаки из НАБОРА2, не превращать
- `-s, -squeeze-repeats` Замещать последовательность знаков, которые повторяются, из перечисленных в последнем НАБОРЕ, на один такой знак
- `-t, -truncate-set1` Сначала сократить НАБОР1 до размеров НАБОРА2

- (j) `uniq` - это утилита, с помощью которой можно вывести или отфильтровать повторяющиеся строки в отсортированном файле.

Синтаксис: `$ uniq опции файл_источник файл_для_записи`

Параметры:

- `-u (-unique)` — выводит исключительно те строки, у которых нет повторов.
- `-d (-repeated)` — если какая-либо строка повторяется несколько раз, она будет выведена лишь единожды.
- `-D` — выводит только повторяющиеся строки.
- `-all-repeated[=МЕТОД]` — то же самое, что и `-D`, но при использовании этой опции между группами из одинаковых строк при выводе будет отображаться пустая строка. [=МЕТОД] может иметь одно из трех значений — `none` (применяется по умолчанию), `separate` или `prepend`.
- `-group[=МЕТОД]` — выводит весь текст, при этом разделяя группы строк пустой строкой. [=МЕТОД] имеет значения `separate` (по умолчанию), `prepend`, `append` и `both`, среди которых нужно выбрать одно.

- (k) `od` - это утилита, которая по умолчанию преобразует входные данные в несколько форматов с восьмеричным форматом.

Синтаксис: `$ od [options] .. [file_name]`

Параметры:

- `-b` - распечатать содержимое файла в восьмеричном формате.
- `-c` - распечатать содержимое файла в символьном формате.
- `-An -c` - распечатать содержимое файла в символьном формате без указания смещения.

- (l) `sum` - это утилита, которая выводит контрольную сумму файла, а также количество блоков в нем.

Синтаксис: `$ sum опции файл`

Параметры:

- `-c` - выполнить проверку по файлу контрольных сумм;
- `-b` - работать в двоичном формате;
- `-t` - работать в текстовом формате;
- `-w` - выводить предупреждения о неверно отформатированном файле сумм;
- `-quiet` - не выводить сообщения об успешных проверках.

- (m) `cat` - эта утилита последовательно выводит файлы, объединяя их в единый поток.

Синтаксис: `$ cat опции файл1 файл2 ...`

Параметры:

- `-b` - нумеровать только непустые строки;
- `-E` - показывать символ `$` в конце каждой строки;
- `-n` - нумеровать все строки;
- `-s` - удалять пустые повторяющиеся строки;
- `-T` - отображать табуляции в виде `Í`;
- `-h` - отобразить справку;
- `-v` - версия утилиты.

- (n) tar - это утилита, с помощью которой реализуется архивирование и разархивирование файлов.
Синтаксис: \$
Параметры:
-
- (o) gzip
Синтаксис: \$
Параметры:
-
- (p) bzip2
Синтаксис: \$ tar опции архив.tar файлы_для_архивации
Для распаковки: \$ tar опции архив.tar
Параметры:
- -C dir -directory=DIR Сменить директорию перед выполнением операции на dir
 - -f file -file Вывести результат в файл (или на устройство) file
 - -j -bzip2 Перенаправить вывод в команду bzip2
 - -p -same-permissions Сохранить все права доступа к файлу
 - -v -verbose Выводить подробную информацию процесса
 - -totals Выводить итоговую информацию завершенного процесса
 - -z -gzip Перенаправить вывод в команду gzip
- (q) head - эта утилита по умолчанию выводит первые 10 строк файла
Синтаксис: \$ head опции файл
Параметры:
- -c (-bytes) — позволяет задавать количество текста не в строках, а в байтах. При записи в виде -bytes=[-]NUM выводит на экран все содержимое файла, кроме NUM байт, расположенных в конце документа.
 - -n (-lines) — показывает заданное количество строк вместо 10, которые выводятся по умолчанию. Если записать эту опцию в виде -lines=[-]NUM, будет показан весь текст кроме последних NUM строк.
 - -q (-quiet, -silent) — выводит только текст, не добавляя к нему название файла.
 - -v (-verbose) — перед текстом выводит название файла.
 - -z (-zero-terminated) — символы перехода на новую строку заменяет символами завершения строк.
- (r) iconv - эта утилита для преобразования текста из одной кодировки в другую.
Синтаксис: \$ iconv -f [исходная кодировка] -t [конечная кодировка] [файл]
Параметры:
- -f - указать исходную кодировку.
 - -t - указать конечную кодировку.
- (s) md5sum - это утилита, которая выводит контрольную сумму файла.
Синтаксис: \$ md5sum опции файл
Параметры:
- -c - выполнить проверку по файлу контрольных сумм;
 - -b - работать в двоичном формате;
 - -t - работать в текстовом формате;
 - -w - выводить предупреждения о неверно отформатированном файле сумм;
 - -quiet - не выводить сообщения об успешных проверках.
- (t) du - эта утилита показывает сколько места занимают папки и файлы на диске.
Синтаксис: \$ du опции /путь/к/папке
Параметры:

- -a, -all - выводить размер для всех файлов, а не только для директорий, по умолчанию размер выводится только для папок;
 - -B, -block-size - указать единицы вывода размера, доступно: K,M,G,T,P,E,Z,Y для 1024 и KB, MB и так далее для 1000;
 - -c, -total - выводить в конце общий размер всех папок;
 - -d, -max-depth - максимальная глубина вложенности директорий;
 - -h, -human-readable - выводить размер в единицах измерения удобных для человека;
 - -inodes - выводить информацию об использовании inode;
 - -L, -dereference - следовать по всем символическим ссылкам;
 - -l, -count-links - учитывать размер файла несколько раз для жестких ссылок;
 - -P, -no-dereference - не следовать по символическим ссылкам, это поведение используется по умолчанию;
 - -S, -separate-dirs - не включать размер подпапок в размер папки;
 - -si - выводить размер файлов и папок в системе si, используется 1000 вместо 1024;
 - -s, -summarize - выводить только общий размер;
 - -t, -threshold - не учитывать файлы и папки с размером меньше указанного;
 - -time - отображать время последней модификации для файла или папки, вместо времени модификации можно выводить такие метки: atime, access, use, ctime;
 - -X, -exclude - исключить файлы из подсчёта;
 - -x, -one-file-system - пропускать примонтированные файловые системы;
 - -version - вывести версию утилиты.
- (u) file - эта утилита указывает тип данных, хранящихся внутри файла.
Синтаксис: \$ file опции название_документа
Параметры:
- -b, -brief — запрет на демонстрацию имен и адресов файлов в выводе команды;
 - -i, -mime — определение MIME-типа документа по его заголовку;
 - -mime-type, -mime-encoding — определение конкретного элемента MIME;
 - -f, -files-from — анализ документов, адреса которых указаны в простом текстовом файле;
 - -l, -list — список паттернов и их длина;
 - -s, -special-files — предотвращение проблем, которые могут возникнуть при чтении утилитой специальных файлов;
 - -P — анализ определенной части файла, которая обозначается различными параметрами;
 - -r, -raw — отказ от вывода /ooo вместо непечатных символов;
 - -z — анализ содержимого сжатых документов.
- (v) touch - эта утилита позволяет создавать пустые файлы
Синтаксис: \$ touch [Опции] Файл
Параметры:
- —help и —version используются редко. Первая опция даст возможность прочесть официальную документацию, а вторая отобразит текущую версию утилиты.
 - -a отвечает за изменение времени доступа к указанному файлу.
 - -m изменяет время модификации.
 - -c определяет, что объект с указанным именем создан не будет.
 - -t позволит использовать время доступа и модификации указанного файла.
 - -t предназначена для изменения даты и времени путем ручного ввода.
 - -d использует дату и время, заданные в виде строки.
- (w) find - эта утилита позволяет найти нужную информацию в системе. Синтаксис: \$ find [папка] [параметры] критерий шаблон [действие]
Параметры:

- -P - никогда не открывать символические ссылки.
- -L - получает информацию о файлах по символическим ссылкам. Важно для дальнейшей обработки, чтобы обрабатывалась не ссылка, а сам файл.
- -maxdepth - максимальная глубина поиска по подкаталогам, для поиска только в текущем каталоге установите 1.
- -depth - искать сначала в текущем каталоге, а потом в подкаталогах.
- -mount искать файлы только в этой файловой системе.
- -version - показать версию утилиты find.
- -print - выводить полные имена файлов.
- -type f - искать только файлы.
- -type d - поиск папки в Linux.

Критерии:

- -name - поиск файлов по имени.
- -perm - поиск файлов в Linux по режиму доступа.
- -user - поиск файлов по владельцу.
- -group - поиск по группе.
- -mtime - поиск по времени модификации файла.
- -atime - поиск файлов по дате последнего чтения.
- -nogroup - поиск файлов, не принадлежащих ни одной группе.
- -nouser - поиск файлов без владельцев.
- -newer - найти файлы новее чем указанный.
- -size - поиск файлов в Linux по их размеру.

(x) mktmp - эта утилита позволяет создавать временные файлы и каталоги.

Синтаксис: \$ mktmp [параметры] имя_файла

Параметры:

- -d - создать временную директорию.
- -suffix= - указать суффикс у имени файла/папки.

Пункты 1-7 отчета составляются строго до начала лабораторной работы. Допущен к выполнению работы. Подпись преподавателя ____

8. Распечатка протокола (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

1. wc test1	Печатает количество строк, слов и байтов в файле
2. wc -c test1	Печатает только количество байтов
3. wc -w test1	Печатает только количество слов
4. wc -l test1	Печатает только количество строк
5. wc -m test1	Печатает только количество символов
6. dd if=20/test1 of=20/test2	Копирует файл в директории 20 test1 в test2 в той же директории
7. diff test1 file3	Построчно сравнивает 2 файла
8. diff -q test1 file3	Печатает только различаются ли 2 файла
9. grep -F sd test1	Ищет и печатает строки файла, в которых содержится паттерн sd, флаг -F показывает, что паттерн - фиксированная строка
10. join file1 file2	Соединяет 2 файла по общему полю, в данном случае - по номеру строки, самому первому полю
11. sort file1	Печатает в стандартный вывод отсортированный по строкам файл
12. tail file1	Печатает последние 10 строк файла
13. tail -n 3 file1	Печатает последние 3 строки файла
14. tee file2	Записывает ст. ввод в файл, одновременно выводит его в ст. вывод
15. tee file2	Добавляет(не перезаписывает) ст. ввод в файл, одновременно выводит его в ст. вывод
16. tr a b	Заменяет символы а в ст. вводе на b и выводит полученный результат в ст. вывод
17. uniq file1	Выводит уникальные строки в ст. вывод
18. uniq file1 file2	Записывает уникальные строки первого файла во второй файл
19. uniq -D file1	Выводит только повторяющиеся строки
20. uniq -d file1	Выводит только повторяющиеся строки по одному разу
21. uniq -all-repeated=separate file1	Выводит только повторяющиеся строки, блоки одинаковых строк разделены пустой строкой
22. uniq -group file1	Выводит все строки в ст. вывод, блоки одинаковых строк разделены пустой строкой
23. od -b file1	Выводит восьмеричные коды каждого символа из файла, начало файла 0000000, конец число символов в файле, также в восьмеричном формате(смещение)
24. od -c file1	Выводит все символы файла, разделенные пробелом
25. od -An -c file1	Выводит все символы файла, разделенные пробелом без смещения
26. od -i file1	Выводит содержимое файла в виде десятичного целого числа
27. od -o file1	Выводит содержимое файла в виде восьмеричных двухбайтовых блоков
28. sum file1	Выводит контрольную сумму файла и количество блоков
29. cut -b 5-20 file4	Вырезает из файла все байты от 5 до 20 и выводит их в ст. вывод
30. tar -cvf arch.tar file1 file2 file3 test1	Создаёт архив arch.tar, архивирует в него 4 файла, и выводит их имя в ст. вывод
31. tar -tf arch.tar	Выводит содержимое архива в ст. вывод
32. tar -xvf arch.tar	Разархивирует архив в текущую директорию
33. gzip -c arch.tar > arch.gz	сжимает архив, результат записывает в arch.gz
34. gzip -cd arch.gz	разжимает архив обратно, выводит результат в ст. вывод
35. bzip2 arch.tar	сжимает архив в архив.bz2, сам архив удаляется.

36. bzip2 -k arch.tar	сжатие с сохранением оригинала
37. head file1	Выводит первые 10 строк файла
38. head -n 2 file1	Выводит первые 2 строки файла
39. iconv -f KOI8-RU -t UTF-8 file1	Переконвертирует файл из KOI8-RU в UTF-8, результат выведет в консоль
40. iconv -f KOI8-RU -t UTF-8 file1 -o file2	Переконвертирует файл из KOI8-RU в UTF-8, результат запишет во второй файл
41. md5sum file2	Выводит контрольную сумму в ст. вывод
42. md5sum file2 > file2.md5	Записывает контрольную сумму в .md5 файл
43. du Labs	Выводит в ст. вывод то, сколько места занимает данная директория и её поддиректории
44. du -a Labs	Выводит в ст. вывод то, сколько места занимает каждый файл в директории
45. du -B K Labs	Выводит в ст. вывод то, сколько места занимает данная директория и её поддиректории в килобайтах
46. du -c Labs	Выводит в ст. вывод то, сколько места занимает данная директория и её поддиректории и общий занимаемый объём
47. du -d 1 Labs	Выводит в ст. вывод то, сколько места занимает данная директория и её поддиректории, с глубиной прохода 1
48. du -h Labs	Выводит в ст. вывод то, сколько места занимает данная директория и её поддиректории, в человеко-читаемом варианте
49. file file1	Выводит в ст. вывод тип данных, которые хранятся внутри file1
50. file -b file1	Выводит в ст. вывод тип данных, которые хранятся внутри file1, но не выводит название файла
51. file -i file1	Выводит в ст. вывод MIME-тип файла по его заголовку
52. file -l file1	Выводит в ст. вывод паттерны файла и их длину.
53. touch emptyfile	Создает пустой файл
54. find . -name "file1"	Ищет file1 в текущей директории
55. find	Ищет все файлы в текущей директории(аналог find . и find . -print)
56. find -type d	Ищет все папки в текущей директории
57. find -type f	Ищет все файлы в текущей директории
58. mktemp tempfileXXX	создает временный файл в текущей директории с именем tempfileПроизвольный набор 4 символов
59. mktemp -d tempdirXXX	создает временный каталог в текущей директории
60. mktemp tempfileXXX -suffix=Alaa	создает в текущей директории временный файл с именем tempfileAlaa

9. Дневник отладки должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.
10. Замечания автора по существу работы: Нет, ссылка на мой гитхаб со всеми лабораторными работами - <https://github.com/Timur-ux/Labs.git>
11. Выводы: В результате работы я получил навыки работы с основными утилитами UNIX для обработки файлов.

Недочёты при выполнении задания могут быть устранены следующим образом: Недочётов нет

Подпись студента: ___