

Лабораторная работа 3

ОБРАБОТКА ТЕКСТОВЫХ ДАННЫХ. ПОЛЬЗОВАТЕЛЬСКОЕ ОКОУРУЖЕНИЕ

Цель работы

Познакомиться с инструментами работы с текстовыми данными доступными в ОС Linux. Получить представление о пользовательском окружении.

1. Теоретические сведения

1.1. Обработка текстовых данных

В операционных системах текстовые файлы используются для хранения данных различного предназначения. В качестве примеров можно выделить файлы конфигурации, системные журналы, файлы с исходным кодом программ и др. Для работы с этими данными разработано большое количество утилит.

Для объединения содержимого нескольких файлов и вывод его в стандартный канал вывода или в файл, используется команда **cat**. Опция **-n** производит нумерацию строк при выводе.

```
$ cat file1 file2 > file3
```

Команда **less** позволяет организовать постраничную работу с большим набором данных.

```
$ less file1
```

Утилита **split** выполняет разделение исходного файла. Например на файлы содержащие определенное количество байт (опция **-b**), строк (опция **-l**).

Утилита **sort** выполняет сортировку поступающих ей на вход данных. Использование дополнительных опций позволяет провести сортировку по одному из полей сгруппированных данных.

```
$ ls -l ~ | sort -n -k 5
```

Приведенная выше команда производит числовую (опция **-n**) сортировку полученных от команды **ls** данных по пятому столбцу (опция **-k**).

Утилита **cut** выполняет фильтрацию текста по столбцам. В качестве опций команда принимает номер поля (**-f**), разделитель (**-d**) и др.

\$ cut -d : -f 1 file1

Приведенная выше команда из множества столбцов разделенных символом «:» в файле **file1** выбирает первый.

Утилита **uniq** удаляет соседние повторяющиеся строки в файле. Опции команды позволяют также найти не уникальные строки и подсчитать количество вхождений каждой строки. Утилиту **uniq** часто используют совместно с утилитой **sort**.

\$ cat filename | sort | uniq

Команды **head** и **tail** используются для отображения выбранного числа строк в начале или в конце файла. По умолчанию число строк равно 10. Изменить количество выводимых строк можно с помощью опции **-n**.

\$ tail -n 3 file1

Команда **wc** используется для подсчета строк, слов, байт и символов в файле. С помощью опций команде можно передать какую конкретно информацию необходимо получить из файла.

Утилита **diff** производит построчное сравнение файлов переданных в качестве аргументов.

\$ diff file1 file2

Помимо утилит командной строки пользователям Linux доступно большое количество текстовых редакторов. Существуют как консольные (**vi**, **nano**, **mcedit**) редакторы, так и редакторы, используемые в графическом режиме (**gedit**, **kate**). Функциональность некоторых из них можно расширить установкой дополнительных модулей-плагинов.

Таблица 1. Примеры команд

Команда	Описание
cat	Вывод объединенного содержимого файла(-ов).
less	Постраничный вывод файла
split	Деление файла на части
sort	Сортировка и/или слияние файлов
cut	Выборка отдельных полей из строк файла

uniq	Выдача информации о повторяющихся строках файла
head	Вывод начальных строк файла
tail	Вывод конечных строк файла
diff	Выявление различий между файлами
wc	Подсчет количества символов, слов и строк в файле

1.2. Среда окружения

При работе с командной оболочкой определен набор переменных, описывающих текущий сеанс взаимодействия пользователя с системой, называемый *окружением* (environment).

Переменные окружения доступны одновременно нескольким процессам.

Загрузка переменных окружения из конфигурационных файлов происходит при запуске командного интерпретатора. В дополнение к переменным окружения, оболочка так же хранит псевдонимы и функции оболочки.

Список всех установленных переменных можно получить, используя команды **env** или **set** без опций и аргументов.

Установка новых и изменение значения существующих переменных среды окружения осуществляется путем *экспортирования* (помещения в среду):

\$ export переменная=значение

Узнать значение конкретной переменной можно также при помощи команды:

\$ echo \$переменная

Чтобы удалить переменную, используется команда **unset**.

Таблица 2. Некоторые стандартные переменные среды окружения

Имя	Значение
UID	Содержит числовой идентификатор текущего пользователя. Инициализируется при запуске оболочки.
HOME	Домашний каталог текущего пользователя.
PATH	Список каталогов, разделённых двоеточием, в которых командная оболочка выполняет поиск файла, в случае если в командене задан его путь.

PS1	Формат строки-приглашения
PWD	Текущий каталог.
TERM	Тип используемого терминала.
HOSTNAME	Сетевое имя компьютера.

2 Порядок выполнения лабораторной работы

1. Прочитайте теоретический материал по лабораторной работе.
2. Ознакомьтесь с работой команд, приведенных в Таблице 1. Изучите для содержащихся в Таблице 1 команд страницы справочного руководства.
3. Отсортируйте вывод команды `ls -l` по дате изменения содержимого по месяцам.
4. Из файла `passwd_example` получите имена всех пользователей, содержащихся в первом поле каждой строки, и поместите отсортированный в обратном порядке результат в файл `cut_result`.
5. С помощью текстового редактора (`gedit`, `pano`) измените имена нескольких пользователей в этом файле и сохраните результат в новый файл `cut_result2`.
6. Сравните содержимое файлов с именами пользователей с помощью программы `diff`.
7. Добавьте к содержимому файла `cut_result` содержимое файла `cut_result2`.
8. С помощью команды `uniq` избавьтесь от дубликатов в файле `cut_result`.
9. С помощью одной команды получите домашний каталог пользователя `user` из файла `passwd_example`.
10. Выведите на экран значения всех переменных среды окружения. Проанализируйте полученные результаты и объясните значения известных вам переменных окружения.
11. Определите тип используемого терминала.
12. Измените содержимое переменной `PS1` так, чтобы в приглашении в качестве скобок использовались символы «< >».
13. Выполните команду `PS1="\[\033[0;41m\]<\u@\h \W>\$\[\033[0m\] "`.

3. Контрольные вопросы

1. Какие утилиты для работы с текстом вы знаете?
2. Какая опция используется для изменения порядка сортировки `sort`?
3. Что такое переменные окружения?
4. Как задать значение переменной окружения и как вывести его на экран?
5. Как изменить значения переменных окружения, удалить переменную?