

Упражнения к теме „Файлы и файловые дескрипторы“

1. Выведите текущий каталог:

```
# pwd
```

2. Перейдите в каталог `/usr/lib` с использованием абсолютного пути:

```
# cd /usr/lib
```

3. Перейдите в каталог `/home/root` с использованием относительного пути:

```
# cd ../../home/root
```

4. Создайте файл `file1`:

```
# touch file1
```

5. Выведите метаданные этого файла:

```
# ls -l file1
```

6. Выведите метаданные с точным временем последнего изменения:

```
# ls --full-time file1
```

7. Повторите команду `touch file1`. Что изменилось в метаданных?

8. Попробуйте запустить файл `file1`:

```
# ./file1
```

Почему ядро отказалось исполнять его?

9. Разрешите исполнение файла для "остальных" пользователей:

```
# chmod 645 ./file1
```

Можно ли его исполнить теперь?

10. Разрешите исполнение файла для владельца файла. Убедитесь в том, что теперь файл исполняется.

11. Выведите метаданные для файлов `/dev/tty1` и `/dev/sda1`. Какой файл является символьным, а какой блочным? Какие старшие и младшие номера устройств?

12. Удалите файл:

```
# rm ./file1
```

13. Выведите список текущих точек монтирования:

```
# mount
```

Какая файловая система (тип и устройство) примонтирована к корню дерева (`/`)?

14. Выведите список запущенных процессов:

```
# ps -A
```

В левом столбце указан номер каждого процесса.

15. Выберите один из процессов и по его номеру найдите соответствующую папку в виртуальной файловой системе `proc`:

```
# ls /proc/<id>
```

16. Выведите имя команды и статус выбранного процесса:

```
# cat /proc/<id>/comm  
# cat /proc/<id>/status
```

В каком состоянии (State) находится процесс?

17. Выведите текст в терминал с помощью команды `echo`:

```
# echo 123
```

18. Выведите текст в терминал с использованием управляющих последовательностей `\b` и `\n`:

```
# echo -e aaa\\nbbb  
# echo -e aaa\\bccc
```

Замечание. `shell` удаляет обратную косую черту из аргументов команды. Чтобы передать одну косую черту в команду `echo`, ее необходимо ввести дважды. В домашнем задании двойную косую черту использовать не нужно.

19. Запишите произвольный текст в файл `students` с помощью перенаправления вывода:

```
# echo aaa > students
```

20. Убедитесь в том, что каждая последующая команда `echo` перезаписывает содержимое файла.

21. Используя управляющую последовательность `\n`, создайте файл `students`, содержащий строки `Masha`, `Sasha`, `Vika`.

22. Выведите содержимое файла:

```
# cat ./students
```

23. Подключите к выводу `cat` программу поиска текста `grep` с помощью конвейера:

```
# cat ./students | grep asha
```

24. Подключите к выводу `grep` потоковый текстовый редактор `sed` и замените слово `Masha` на `Dasha`:

```
# cat ./students | sed s/Masha/Dasha/g
```

25. Используя утилиты-фильтры `grep` и `sed`, создайте на основе `students` новый файл `students2`, содержащий только имена с суффиксом

„asha” и в котором маленькие буквы „a” заменены на большие „A”.

26.Удалите созданные файлы:

```
# rm ./students  
# rm ./students2
```

27.Перейдите в каталог /dev. Далее используя shell-шаблоны, выведите список файлов с префиксом tty:

```
# ls tty*
```

28.Объясните вывод следующих команд:

```
# ls tty4*  
# ls *4*  
# ls tty49*  
# ls tty49?  
# ls tty4?
```