

## Тесты для shell

- `cd ..`  
`pwd`

Перейти в родительскую директорию, напечатать ее название

- `echo homedir: $HOME username: $USER`  
Напечатать имя домашней директории и имя пользователя

- `cd`  
`pwd`

Перейти по умолчанию в домашнюю директорию. Убедиться, что попали туда

- `pwd > fpwd`  
имя текущей директории записывается в файл `fpwd`
- `who >> fpwd`  
имя пользователя добавляется в конец файла `fpwd`
- `cat < file.in > file.out`

Копирует содержимое файла `file.in` в `file.out`.

- `cat < file.in >> file.out`

Дописывает содержимое файла `file.in` в конец `file.out`.

- `zcat < file.tar.gz | tar -xv`

Распаковывает `.tar.gz` архив.

- `cat < /dev/urandom | head -c 4096 > "file.bin"`

Создаёт файл случайных чисел размером 4096 байт.

- `yes | head`

Печатает 10 раз букву `y`.

- `yes | yes | yes | yes | yes | yes | yes | yes | yes | head`

Печатает 10 раз букву `y`. Проследить с помощью `ps` или `top` из другого окна `xterm`, чтобы не оставались зомби

- `cat < /dev/null | head | head | head | head | head`

Ничего не печатает. Проследить с помощью `ps` или `top` из другого окна `xterm`, чтобы не оставались зомби

- `pwd > current_dir.txt`

Печатает текущую директорию в файл `current_dir.txt`.

- `pwd | cat >> my_home.txt`

Дописывает домашнюю директорию пользователя в файл `my_home.txt`.

- `ls | cat | cat | cat | cat`

Печатает список файлов текущей директории на стандартный вывод.

- `who | cat > who_am_i.txt | cat | cat | cat > who_am_i_not.txt`

Печатает имя пользователя в файл `who_am_i.txt`. Файл `who_am_i_not.txt` создаётся и оказывается пустым.

- `yes | yes | yes | yes | sleep 10 | pwd`

Печатает текущую директорию. Проследить, чтобы приглашение к следующей команде появилось не раньше, чем через 10 сек.

Выводит пронумерованные строки, введенные с клавиатуры, и отсортированные в обратном порядке

- `cat | sort -r | cat -n`

Печатает вводимые строки, отсортированные в обратном порядке и пронумерованные

- `sleep 10; pwd`

Директория печатается через 10 сек

- `sleep 10 & pwd`

Директория печатается сразу

Проверить, что не остается зомби от фонового процесса `sleep 10 &`

- `sleep 5 & sleep 5 & sleep 5 & sleep 5 & sleep 5 &`  
Проверить зомби (с помощью команды `ps` или `top`)

- `echo a b c > f; cat f & ls`

- `echo a b c > f; echo >> f; cat <f`

- `(ls; pwd; who) | cat -n`

- `(((((ls) | cat -n ) | cat -n) | sort -r)`

- `(((((ls) | cat -n ) | cat -n) | sort -r) &`

Проверить зомби

- `(pwd|(cat -n|(cat |(cat) | sort -r ) | cat -n) |cat)`

- `(pwd|(cat -n|(cat |(cat))))`

- `cat f; date; pwd > zz`

- `date; (ls -l |cat -n)>f & pwd`

- `ls && who && pwd`

- `ls || who || pwd`

- `( ls || who || pwd) |cat | sort -r`

- `( ls && who || pwd) |cat -n | sort`

- `pwd && sleep 5 &`

- `ls || who | cat`

- `who |cat || ls`

- `cat nosuchname || pwd`

- `cat nosuchname && pwd`

- `(ls& ) | sleep 10; pwd`

С помощью запуска в bash команды

```
yes "любая_из_тестовых_команд_приведенных_выше" | my_shell
```

контролировать утечку памяти, наблюдая за исполнением из другого терминального окна с помощью команды `top`. Ей следует указать параметр, позволяющий показывать только пользовательские процессы, чтобы системные процессы не мешали наблюдать. После выполнения очередной команды `my_shell` должен освобождать все вспомогательные структуры данных, так что объем используемой памяти не должен расти со временем.