

## Упражнения к теме „Процессы ОС Linux”

1. Скопируйте каталог class-task-templates в общую папку виртуальной машины.
2. Написать программу fork-process, которая создает дочерний процесс и дожидается его завершения. Дочерний процесс выводит свой PID и завершается с кодом 5. Родительский процесс выводит свой PID, а также причину завершения дочернего процесса, код завершения или сигнала, приведшего к завершению.

3. Запустите fork-process:

```
# ./fork-process
parent PID: 9108
child PID: 9109
Child exited with status 5
```

4. Увеличьте время жизни дочернего процесса с помощью функции sleep(). Запустите fork-process и завершите его с помощью сигнала SIGTERM, указав соответствующий PID процесса:

```
# ./fork-process &
# kill <PID>
Child was terminated by signal 15
```

5. Напишите программу catch-signal, которая перехватывает сигналы SIGTERM, SIGKILL, SIGINT, SIGCHLD. В функции main() программа реализует бесконечный цикл чтения из терминала, который прерывается доставкой сигнала. После каждого прерывания программа выводит имена сигналов, прервавших программу.
6. Запустите программу catch-signal и отправьте ей сигнал SIGINT с помощью комбинации Ctrl+C:

```
# ./catch-signal
^C
```

```
Interrupted: sigterm 0, sigkill 0, sigint 1, sigchld 0
```

7. Завершите `catch-signal` из другого терминала:

```
# pkill -9 catch-signal
```

8. Запустите программу `catch-signal` и отправьте ей сигналы `SIGTERM` и `SIGCHLD`:

```
# ./catch-signal &  
# pkill -SIGTERM catch-signal  
Interrupted: sigterm 1, sigkill 0, sigint 0, sigchld 0
```

9. Удалите каталог `class-task-templates` из общей папки.