

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине «Операционная система Linux»

Процессы в операционной системе Linux

Студент

Лобов М.Ю.

Группа АИ-18

Руководитель

Кургасов В.В.

Доцент, к.п.н.

Липецк 2020 г.

Оглавление

Цель работы	3
Задание кафедры.....	4
Часть I	6
Часть II.....	12
Часть III	16
Вывод.....	18

Цель работы

Ознакомиться на практике с понятием процесса в операционной системе. Приобрести опыт и навыки управления процессами в операционной системе Linux.

Задание кафедры

Часть I (выполняют все).

- 1) Загрузиться не root, а пользователем.
- 2) Найти файл с образом ядра. Выяснить по имени файла номер версии Linux.
- 3) Посмотреть процессы `ps -f`. Прокомментировать. Для этого почитать `man ps`.
- 4) Написать с помощью редактора `vi` два сценария `loop` и `loop2`. Текст сценариев:
Loop:

```
while true; do true; done
```


Loop2:

```
while true; do true; echo 'Hello'; done
```
- 5) Запустить `loop2` на переднем плане: `sh loop2`.
- 6) Остановить, послав сигнал `STOP`.
- 7) Посмотреть последовательно несколько раз `ps -f`. Записать сообщение, объяснить.
- 8) Убить процесс `loop2`, послав сигнал `kill -9 PID`. Записать сообщение. Прокомментировать.
- 9) Запустить в фоне процесс `loop`: `sh loop&`. Не останавливая, посмотреть несколько раз: `ps -f`. Записать значение, объяснить.
- 10) Завершить процесс `loop` командой `kill -15 PID`. Записать сообщение, прокомментировать.
- 11) Третий раз запустить в фоне. Не останавливая убить командой `kill -9 PID`.
- 12) Запустить еще один экземпляр оболочки: `bash`.
- 13) Запустить несколько процессов в фоне. Останавливать их и снова запускать. Записать результаты просмотра командой `ps -f`.

Часть II (выполняют все).

1. Запустить в консоли на выполнение три задачи, две в интерактивном режиме, одну - в фоновом.
2. Перевести одну из задач, выполняющихся в интерактивном режиме, в фоновый режим.
3. Провести эксперименты по переводу задач из фонового режима в интерактивный и наоборот.
4. Создать именованный канал для архивирования и осуществить передачу в канал списка файлов домашнего каталога вместе с подкаталогами (ключ -R), одного каталога вместе с файлами и подкаталогами.

Часть III. Индивидуальные задания.

Вариант 4.

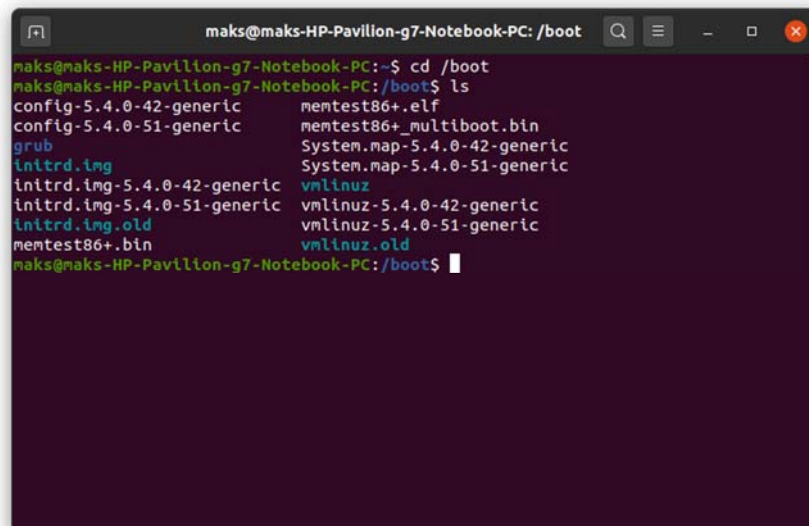
1. Сгенерировать следующую информацию — полный листинг в длинном формате о процессах текущего пользователя: PID, PPID, выделенное время ЦП, время запуска, размер образа.
2. С помощью сигнала SIGTSTP (используя комбинацию клавиш и команду kill) приостановить выполнение процесса, владельцем которого является текущий пользователь. Через несколько секунд возобновить выполнение процесса.
3. Определить идентификатор и имя процесса, созданного последним пользователем root.

Ход работы

Часть I

Для начала работы нам нужно загрузиться не root, а пользователем. Для этого просто запустим терминал с помощью комбинации клавиш CTRL + ALT + T.

Затем найдём файл ядра в директории boot:

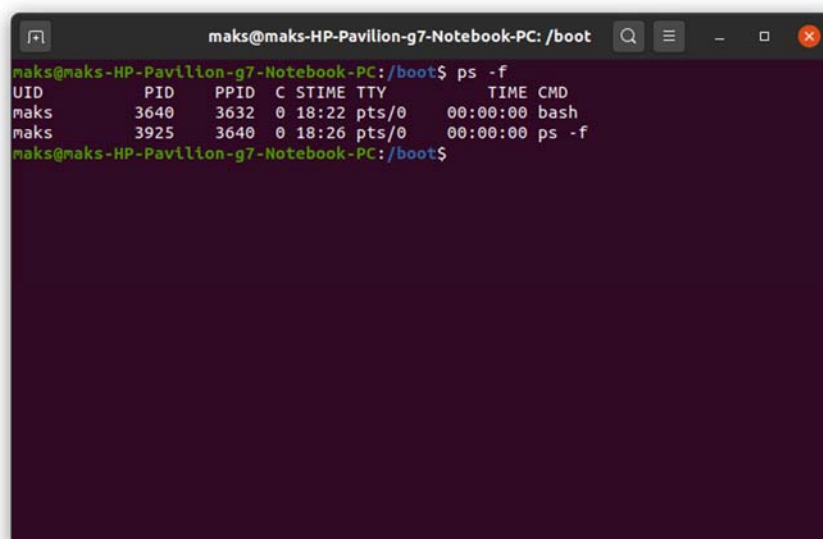


```
maks@maks-HP-Pavilion-g7-Notebook-PC: /boot
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ cd /boot
maks@maks-HP-Pavilion-g7-Notebook-PC:/boot$ ls
config-5.4.0-42-generic      memtest86+.elf
config-5.4.0-51-generic      memtest86+_multiboot.bin
grub                         System.map-5.4.0-42-generic
initrd.img                  System.map-5.4.0-51-generic
initrd.img-5.4.0-42-generic  vmlinuz
initrd.img-5.4.0-51-generic  vmlinuz-5.4.0-42-generic
initrd.img.old              vmlinuz-5.4.0-51-generic
memtest86+.bin              vmlinuz.old
maks@maks-HP-Pavilion-g7-Notebook-PC:/boot$
```

Рисунок 1 – Версия ядра

По рисунку 1 видно (цифры в названии файла vmlinuz), что ядро имеет версию 5.4.0.

Затем посмотрим запущенные процессы с помощью команды ps -f:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: /boot
maks@maks-HP-Pavilion-g7-Notebook-PC:/boot$ ps -f
UID        PID     PPID  C  STIME TTY          TIME CMD
maks       3640     3632  0   18:22 pts/0        00:00:00 bash
maks       3925     3640  0   18:26 pts/0        00:00:00 ps -f
maks@maks-HP-Pavilion-g7-Notebook-PC:/boot$
```

Рисунок 2 – Запущенные процессы

Поясним содержание рисунка 2:

UID - имя пользователя, от имени которого работает процесс;

PID - идентификатор пользователя;

PPID - идентификатор родительского процесса пользователя;

C - расходование ресурсов процессора, в процентах;

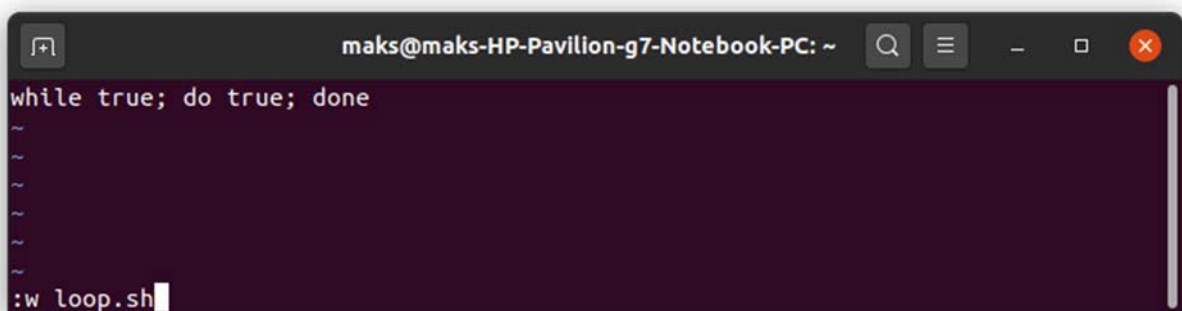
STIME - время, когда процесс был запущен;

TTY - если процесс привязан к терминалу, то здесь будет выведен его номер;

TIME - общее время выполнения процесса (user + system);

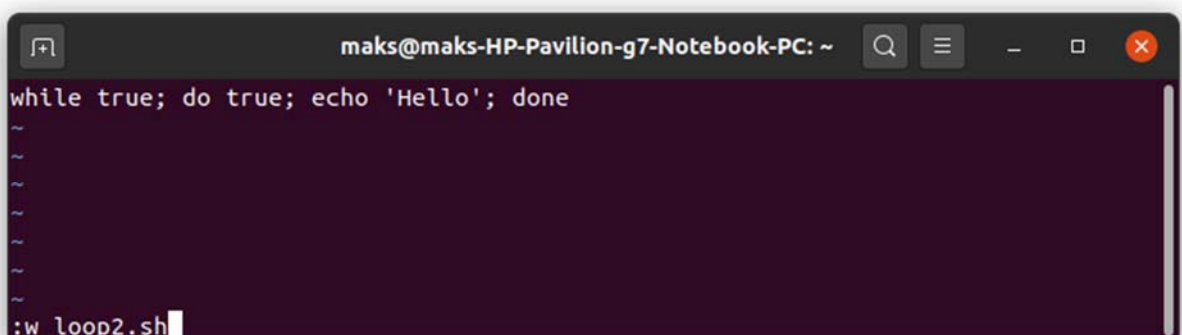
CMD - команда, которой был запущен процесс, если программа не может прочитать аргументы процесса, он будет выведен в квадратных скобках;

Теперь напишем в редакторе Vi два сценария loop и loop2 и сохраним их:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~
while true; do true; done
~
~
~
~
:w loop.sh
```

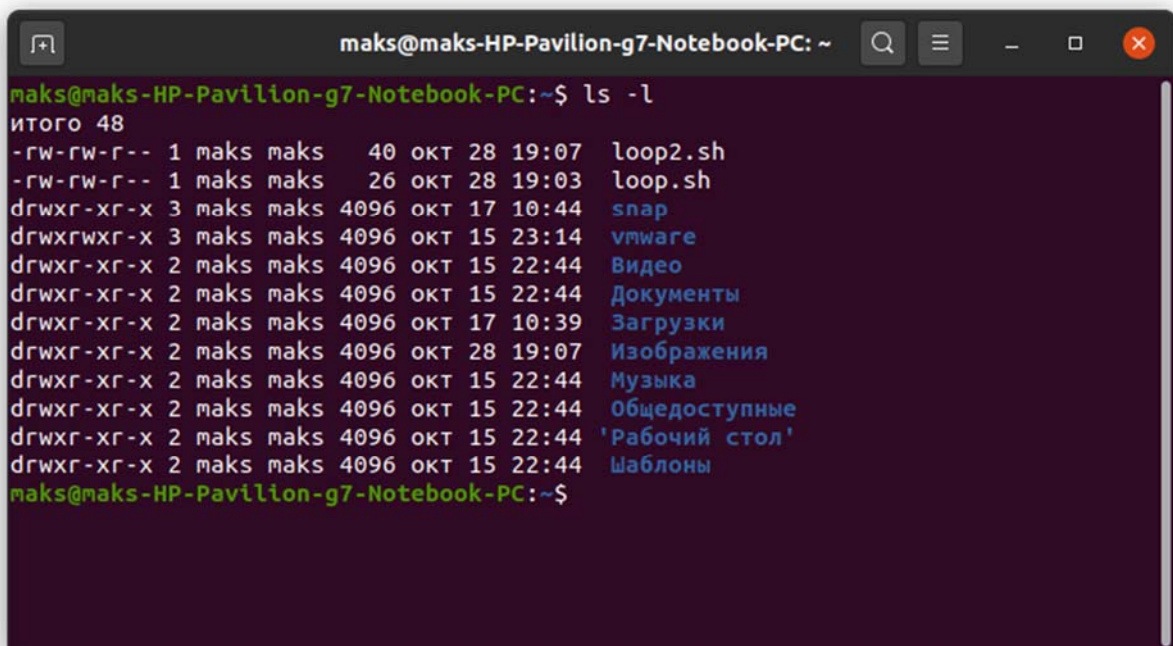
Рисунок 3 – Сценарий loop



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~
while true; do true; echo 'Hello'; done
~
~
~
~
:w loop2.sh
```

Рисунок 4 – Сценарий loop2

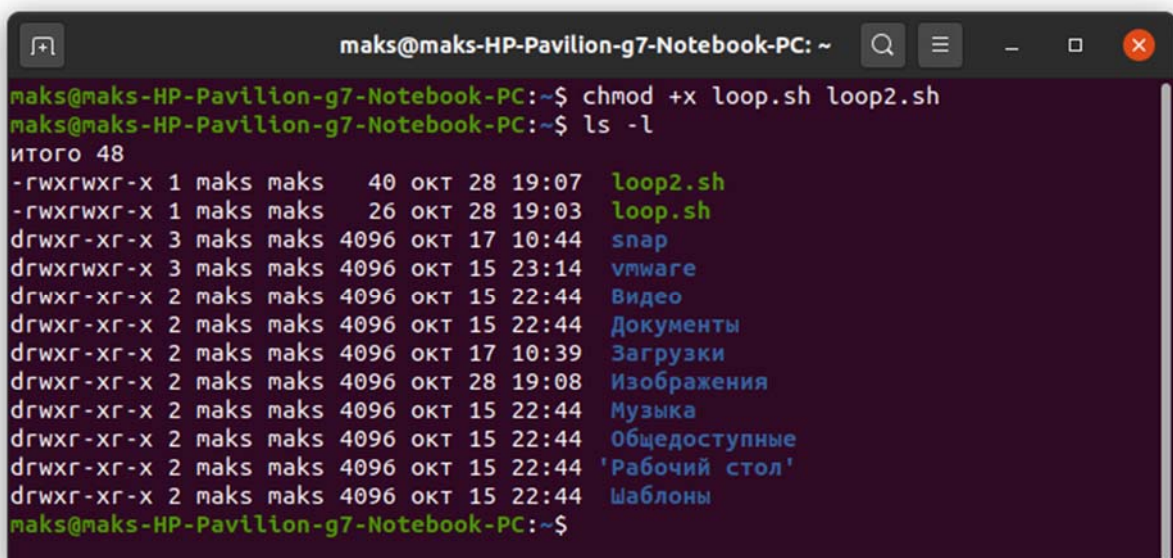
Проверим каталог на наличие созданных файлов:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ls -l  
итого 48  
-rw-rw-r-- 1 maks maks 40 окт 28 19:07 loop2.sh  
-rw-rw-r-- 1 maks maks 26 окт 28 19:03 loop.sh  
drwxr-xr-x 3 maks maks 4096 окт 17 10:44 snap  
drwxrwxr-x 3 maks maks 4096 окт 15 23:14 vmware  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Видео  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Документы  
drwxr-xr-x 2 maks maks 4096 окт 17 10:39 Загрузки  
drwxr-xr-x 2 maks maks 4096 окт 28 19:07 Изображения  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Музыка  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Общедоступные  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 'Рабочий стол'  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Шаблоны  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 5 – Каталог с файлами loop.sh и loop2.sh

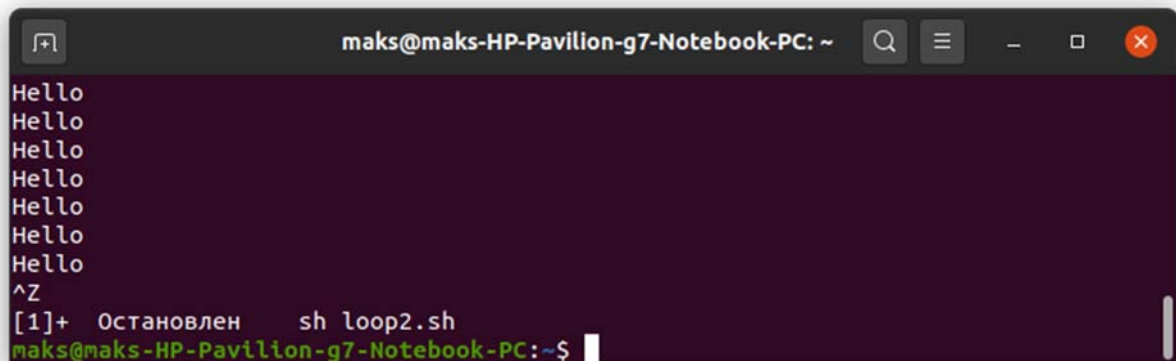
Далее следует изменить права на файлы loop.sh и loop2.sh и добавим право на исполнение с помощью команды chmod:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ chmod +x loop.sh loop2.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ls -l  
итого 48  
-rwxrwxr-x 1 maks maks 40 окт 28 19:07 loop2.sh  
-rwxrwxr-x 1 maks maks 26 окт 28 19:03 loop.sh  
drwxr-xr-x 3 maks maks 4096 окт 17 10:44 snap  
drwxrwxr-x 3 maks maks 4096 окт 15 23:14 vmware  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Видео  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Документы  
drwxr-xr-x 2 maks maks 4096 окт 17 10:39 Загрузки  
drwxr-xr-x 2 maks maks 4096 окт 28 19:08 Изображения  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Музыка  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Общедоступные  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 'Рабочий стол'  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Шаблоны  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

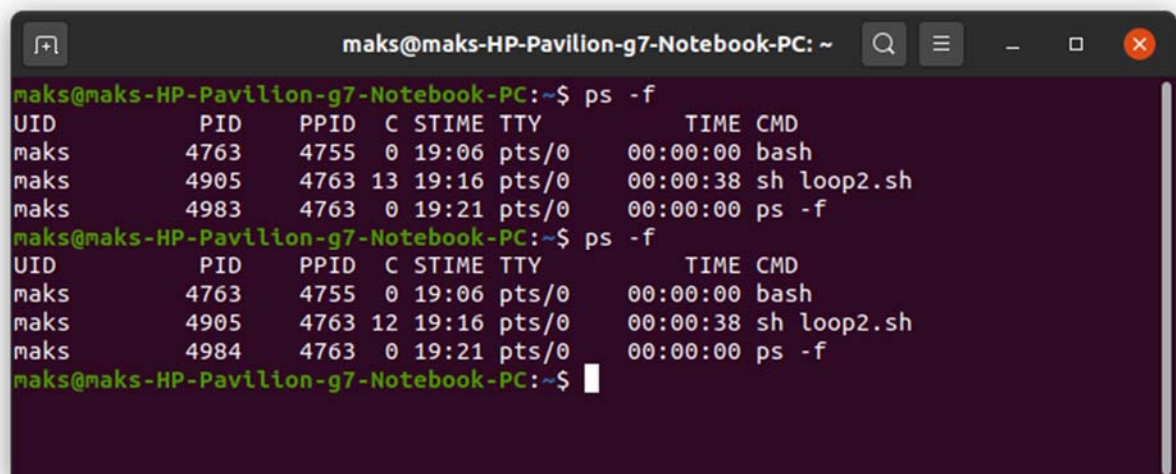
Рисунок 6 – Изменение прав на файлы

Теперь мы можем запустить сценарий на исполнение в интерактивном режиме с помощью команды `sh`. Сделаем это, а также остановим процесс с помощью комбинации клавиш `CTRL + Z`:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
Hello  
Hello  
Hello  
Hello  
Hello  
Hello  
Hello  
^Z  
[1]+  Остановлен    sh loop2.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 7 – Запуск сценария на исполнение и остановка процесса

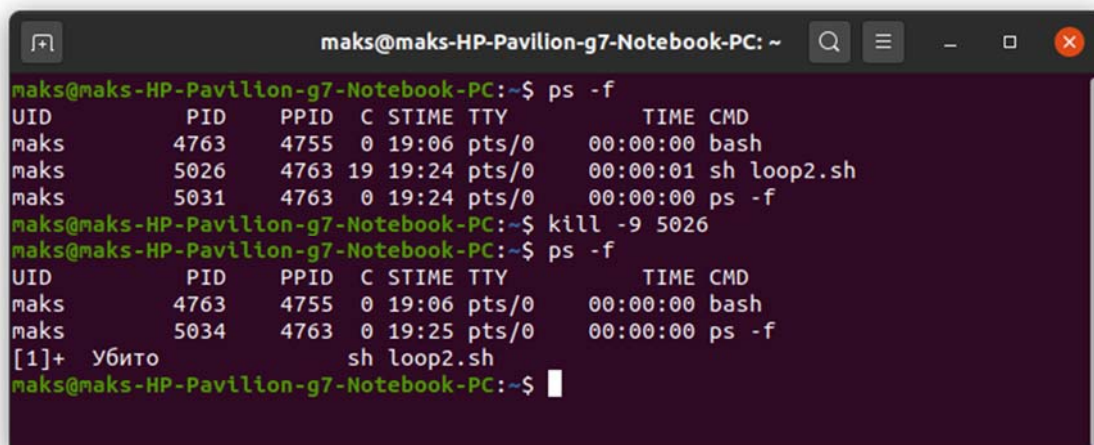


```
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID        PID     PPID  C  STIME TTY          TIME CMD  
maks       4763    4755  0  19:06 pts/0        00:00:00 bash  
maks       4905    4763 13  19:16 pts/0        00:00:38 sh loop2.sh  
maks       4983    4763  0  19:21 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID        PID     PPID  C  STIME TTY          TIME CMD  
maks       4763    4755  0  19:06 pts/0        00:00:00 bash  
maks       4905    4763 12  19:16 pts/0        00:00:38 sh loop2.sh  
maks       4984    4763  0  19:21 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 8 – Два листинга процесса подряд

Отсюда видим, что ресурс процессора, затраченный на процесс `loop2`, уменьшается, что говорит о приостановке процессора.

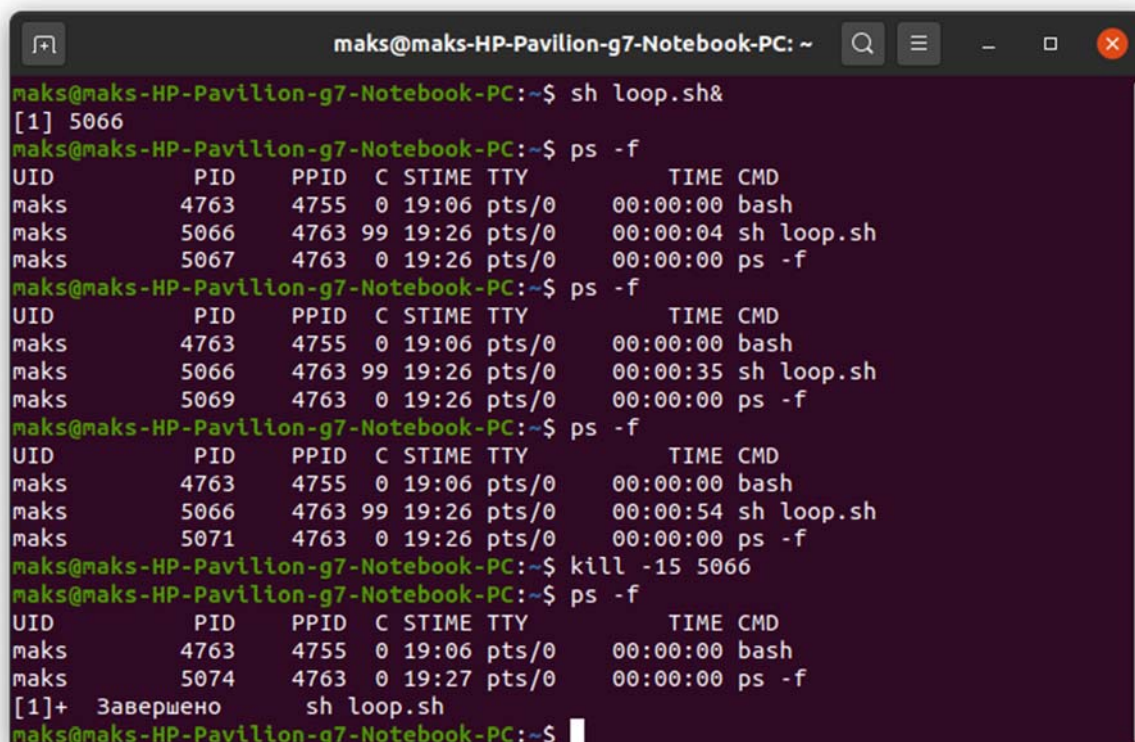
Теперь «убьём» процесс с помощью команды `kill -9` и в качестве аргумента передадим идентификатор процесса (PID):



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5026      4763  19  19:24 pts/0        00:00:01 sh loop2.sh  
maks     5031      4763  0  19:24 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ kill -9 5026  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5034      4763  0  19:25 pts/0        00:00:00 ps -f  
[1]+  Убито                  sh loop2.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 9 – Уничтожение процесса loop2.sh

Теперь запустим сценарий `loop.sh` на исполнение в фоновом режиме с помощью команды `sh loop.sh&`:

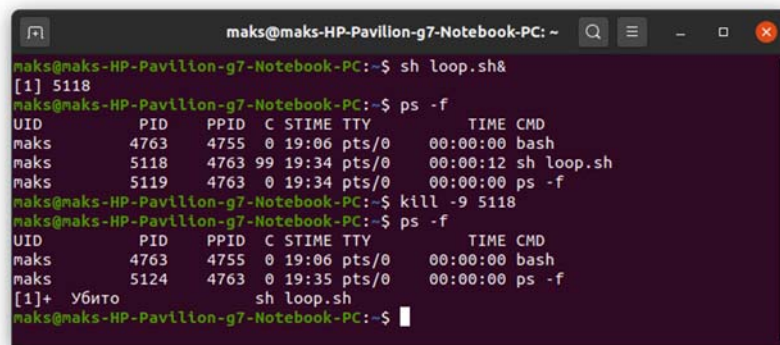


```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh&  
[1] 5066  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5066      4763  99  19:26 pts/0        00:00:04 sh loop.sh  
maks     5067      4763  0  19:26 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5066      4763  99  19:26 pts/0        00:00:35 sh loop.sh  
maks     5069      4763  0  19:26 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5066      4763  99  19:26 pts/0        00:00:54 sh loop.sh  
maks     5071      4763  0  19:26 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ kill -15 5066  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks     4763      4755  0  19:06 pts/0        00:00:00 bash  
maks     5074      4763  0  19:27 pts/0        00:00:00 ps -f  
[1]+  Завершено              sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 10 – Запуск процесса loop.sh в фоновом режиме

По рисунку 10 видно, что ресурс процессора, затраченный на процесс, не уменьшается, что говорит о том, что процесс запущен.

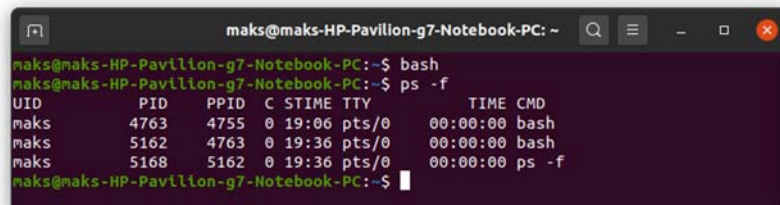
Теперь запустим процесс loop.sh снова и «убьём» его командой kill -9:



```
maks@maks-HP-Pavillon-g7-Notebook-PC: ~  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ sh loop.sh&  
[1] 5118  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks    4763    4755  0  19:06 pts/0        00:00:00 bash  
maks    5118    4763  99  19:34 pts/0        00:00:12 sh loop.sh  
maks    5119    4763  0  19:34 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ kill -9 5118  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks    4763    4755  0  19:06 pts/0        00:00:00 bash  
maks    5124    4763  0  19:35 pts/0        00:00:00 ps -f  
[1]+  Убито                sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$
```

Рисунок 11 – Запуск и убийство процесса loop.sh

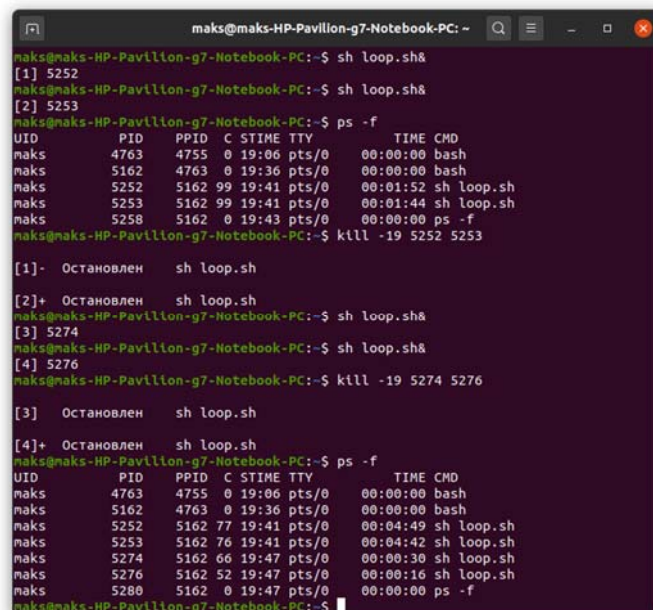
Следующим заданием будет запуск ещё одного экземпляра оболочки с помощью команды bash:



```
maks@maks-HP-Pavillon-g7-Notebook-PC: ~  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ bash  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks    4763    4755  0  19:06 pts/0        00:00:00 bash  
maks    5162    4763  0  19:36 pts/0        00:00:00 bash  
maks    5168    5162  0  19:36 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$
```

Рисунок 12 – Запуск экземпляра оболочки

Теперь запустим несколько процессов в фоне. Будем их останавливать и снова запускать:



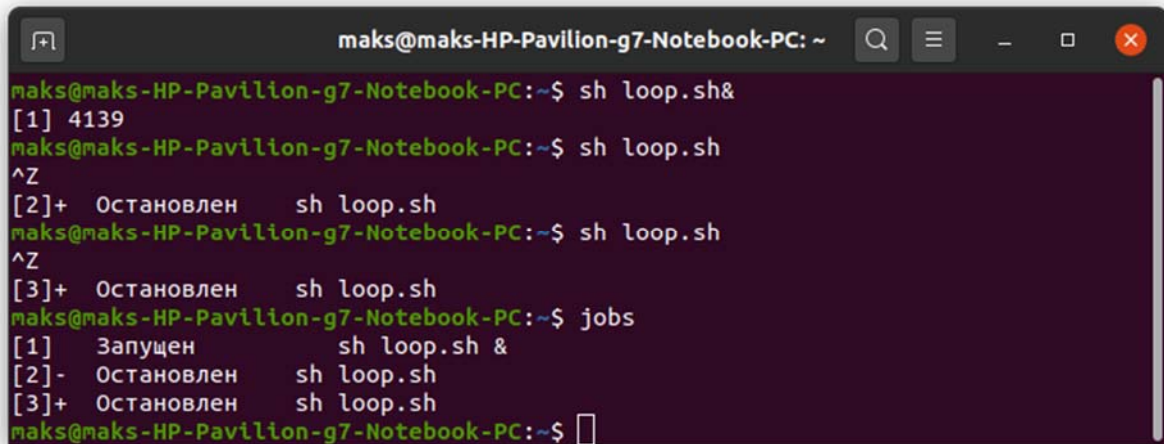
```
maks@maks-HP-Pavillon-g7-Notebook-PC: ~  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ sh loop.sh&  
[1] 5252  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ sh loop.sh&  
[2] 5253  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks    4763    4755  0  19:06 pts/0        00:00:00 bash  
maks    5162    4763  0  19:36 pts/0        00:00:00 bash  
maks    5252    5162  99  19:41 pts/0        00:01:52 sh loop.sh  
maks    5253    5162  99  19:41 pts/0        00:01:44 sh loop.sh  
maks    5258    5162  0  19:43 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ kill -19 5252 5253  
[1]-  Остановлен          sh loop.sh  
[2]+  Остановлен          sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ sh loop.sh&  
[3] 5274  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ sh loop.sh&  
[4] 5276  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ kill -19 5274 5276  
[3]-  Остановлен          sh loop.sh  
[4]+  Остановлен          sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ps -f  
UID      PID     PPID  C  STIME TTY          TIME CMD  
maks    4763    4755  0  19:06 pts/0        00:00:00 bash  
maks    5162    4763  0  19:36 pts/0        00:00:00 bash  
maks    5252    5162  77  19:41 pts/0        00:04:49 sh loop.sh  
maks    5253    5162  76  19:41 pts/0        00:04:42 sh loop.sh  
maks    5274    5162  66  19:47 pts/0        00:00:30 sh loop.sh  
maks    5276    5162  52  19:47 pts/0        00:00:16 sh loop.sh  
maks    5280    5162  0  19:47 pts/0        00:00:00 ps -f  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$
```

Рисунок 13 – Запуск и остановка процессов несколько раз

На этом задания I части выполнены. Перейдём ко II части.

Часть II

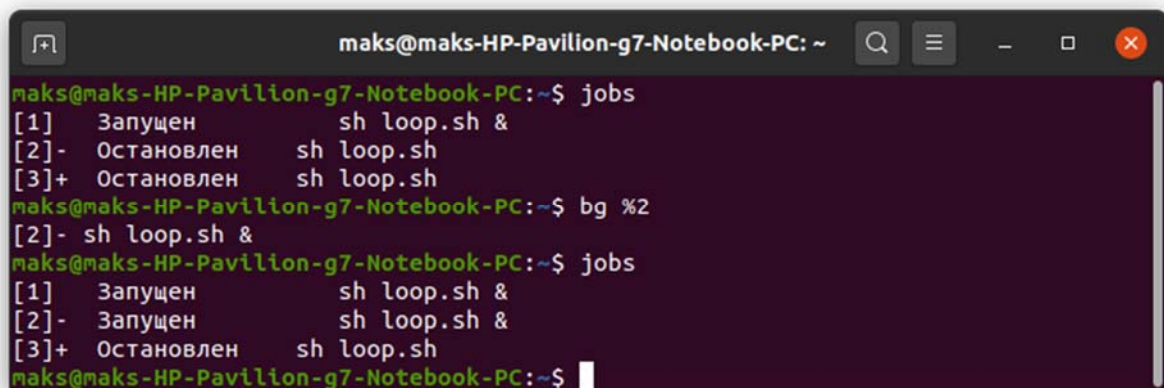
Первым заданием II части будет запуск трёх задач: двух в интерактивном режиме и одной – в фоновом. Затем выведем процессы с помощью команды `jobs`:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh&  
[1] 4139  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh  
^Z  
[2]+ Остановлен sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh  
^Z  
[3]+ Остановлен sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ jobs  
[1] Запущен sh loop.sh &  
[2]- Остановлен sh loop.sh  
[3]+ Остановлен sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 14 – Запуск трёх задач

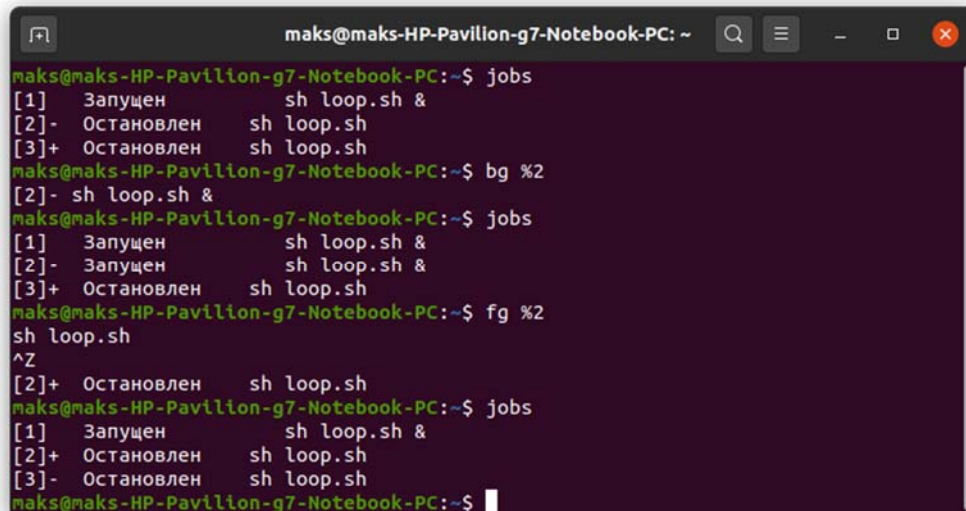
Теперь переведём одну из задач, выполняющихся в интерактивном режиме, в фоновый режим с помощью команды `bg`:



```
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ jobs  
[1] Запущен sh loop.sh &  
[2]- Остановлен sh loop.sh  
[3]+ Остановлен sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ bg %2  
[2]- sh loop.sh &  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ jobs  
[1] Запущен sh loop.sh &  
[2]- Запущен sh loop.sh &  
[3]+ Остановлен sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 15 – Перевод процесса в фоновый режим

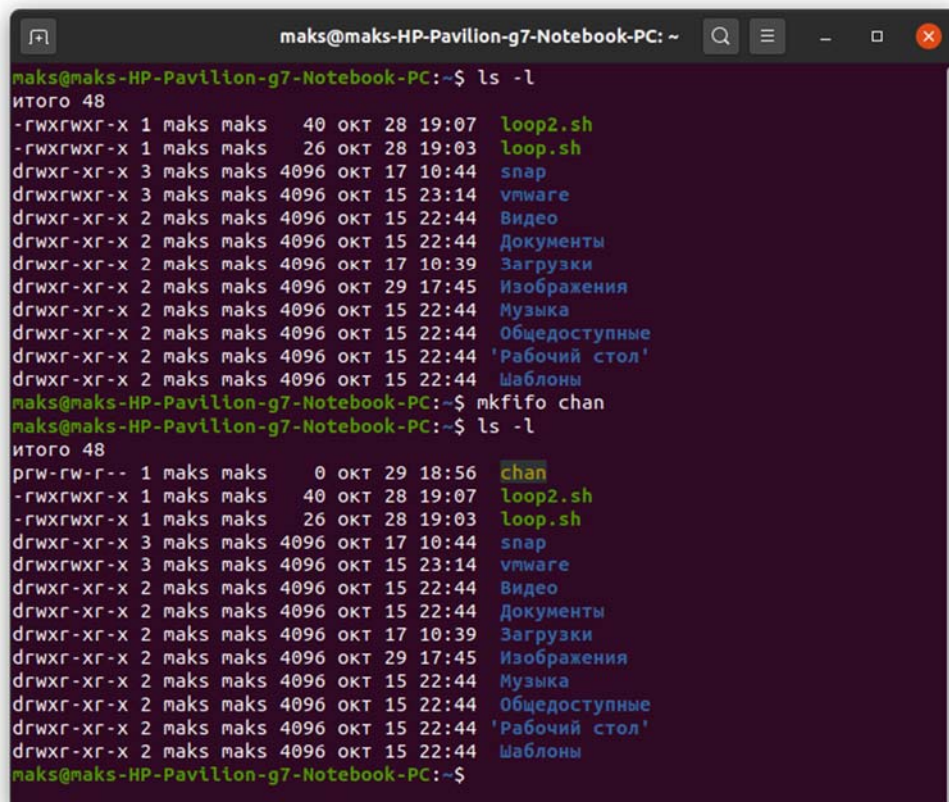
Попробуем переводить процессы в фоновый режим из интерактивного и наоборот – из фонового в интерактивный:



```
maks@maks-HP-Pavillon-g7-Notebook-PC: ~  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ jobs  
[1]  Запущен      sh loop.sh &  
[2]-  Остановлен  sh loop.sh  
[3]+  Остановлен  sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ bg %2  
[2]-  sh loop.sh &  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ jobs  
[1]  Запущен      sh loop.sh &  
[2]-  Запущен      sh loop.sh &  
[3]+  Остановлен  sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ fg %2  
sh loop.sh  
^Z  
[2]+  Остановлен  sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ jobs  
[1]  Запущен      sh loop.sh &  
[2]+  Остановлен  sh loop.sh  
[3]-  Остановлен  sh loop.sh  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$
```

Рисунок 16 – Перевод задачи из фонового режима в интерактивный и наоборот

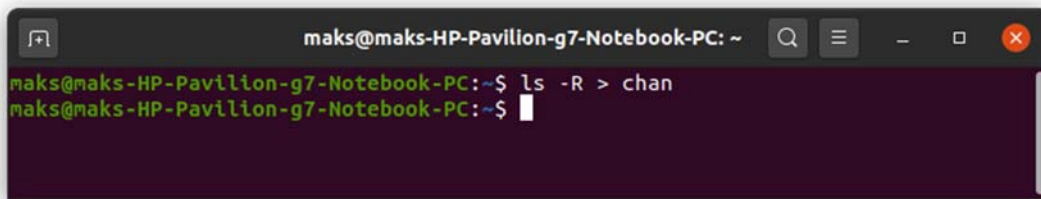
Теперь создадим именованный канал для архивирования с помощью команды `mkfifo`:



```
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ls -l  
итого 48  
-rwxrwxr-x 1 maks maks 40 окт 28 19:07 loop2.sh  
-rwxrwxr-x 1 maks maks 26 окт 28 19:03 loop.sh  
drwxr-xr-x 3 maks maks 4096 окт 17 10:44 snap  
drwxrwxr-x 3 maks maks 4096 окт 15 23:14 vmware  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Видео  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Документы  
drwxr-xr-x 2 maks maks 4096 окт 17 10:39 Загрузки  
drwxr-xr-x 2 maks maks 4096 окт 29 17:45 Изображения  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Музыка  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Общедоступные  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 'Рабочий стол'  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Шаблоны  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ mkfifo chan  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$ ls -l  
итого 48  
rwx-rw-r-- 1 maks maks 0 окт 29 18:56 chan  
-rwxrwxr-x 1 maks maks 40 окт 28 19:07 loop2.sh  
-rwxrwxr-x 1 maks maks 26 окт 28 19:03 loop.sh  
drwxr-xr-x 3 maks maks 4096 окт 17 10:44 snap  
drwxrwxr-x 3 maks maks 4096 окт 15 23:14 vmware  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Видео  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Документы  
drwxr-xr-x 2 maks maks 4096 окт 17 10:39 Загрузки  
drwxr-xr-x 2 maks maks 4096 окт 29 17:45 Изображения  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Музыка  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Общедоступные  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 'Рабочий стол'  
drwxr-xr-x 2 maks maks 4096 окт 15 22:44 Шаблоны  
maks@maks-HP-Pavillon-g7-Notebook-PC:~$
```

Рисунок 17 – Создание именованного канала

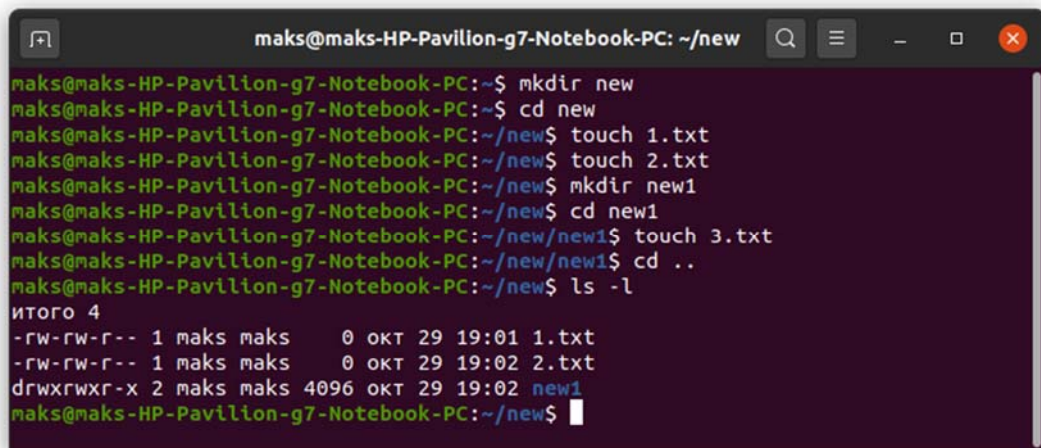
Затем занесём в канал листинг домашнего каталога:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ls -R > chan  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 18 – Заносим в канал листинг домашнего каталога

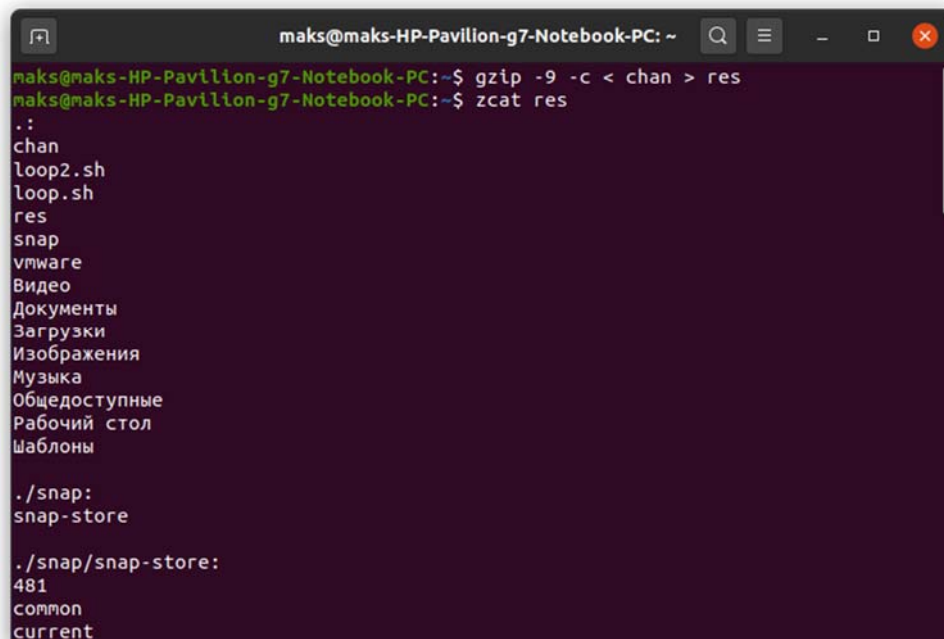
Создадим директорию:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~/new  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ mkdir new  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ cd new  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$ touch 1.txt  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$ touch 2.txt  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$ mkdir new1  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$ cd new1  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new/new1$ touch 3.txt  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new/new1$ cd ..  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$ ls -l  
итого 4  
-rw-rw-r-- 1 maks maks    0 окт 29 19:01 1.txt  
-rw-rw-r-- 1 maks maks    0 окт 29 19:02 2.txt  
drwxrwxr-x 2 maks maks 4096 окт 29 19:02 new1  
maks@maks-HP-Pavilion-g7-Notebook-PC:~/new$
```

Рисунок 19 – Создание нового каталога

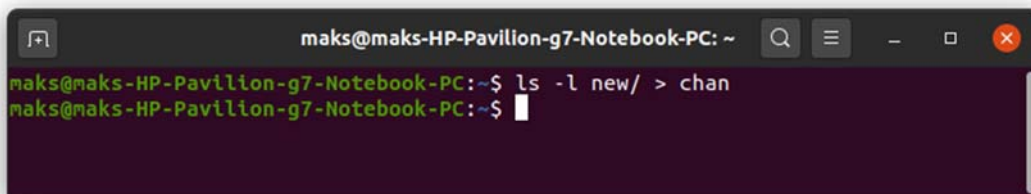
Откроем листинг домашнего каталога:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ gzip -9 -c < chan > res  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ zcat res  
.:  
chan  
loop2.sh  
loop.sh  
res  
snap  
vmware  
Видео  
Документы  
Загрузки  
Изображения  
Музыка  
Общедоступные  
Рабочий стол  
Шаблоны  
  
./snap:  
snap-store  
  
./snap/snap-store:  
481  
common  
current
```

Рисунок 20 – Листинг домашнего каталога

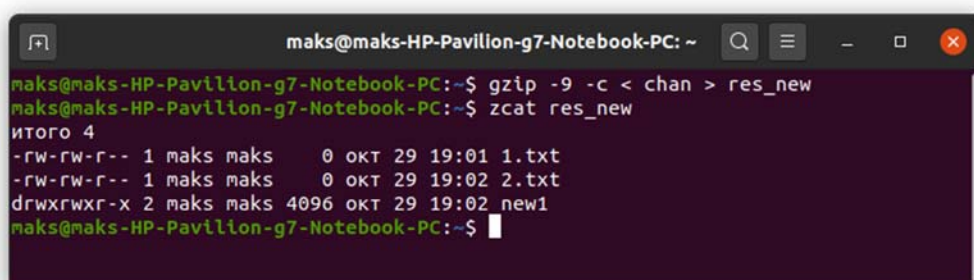
Теперь занесём в канал листинг созданного каталога:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ls -l new/ > chan  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 21 – Заносим в канал листинг созданного каталога

Теперь архивируем содержимое каталога и откроем архив:



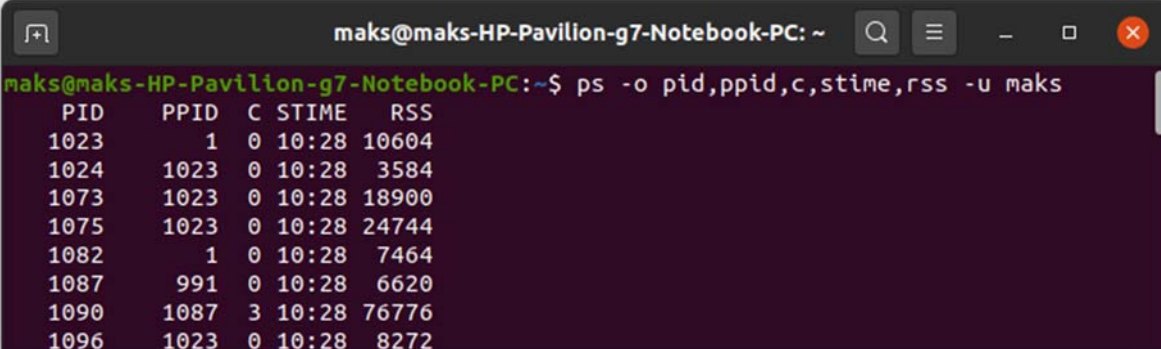
```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ gzip -9 -c < chan > res_new  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ zcat res_new  
итого 4  
-rw-rw-r-- 1 maks maks    0 окт 29 19:01 1.txt  
-rw-rw-r-- 1 maks maks    0 окт 29 19:02 2.txt  
drwxrwxr-x 2 maks maks 4096 окт 29 19:02 new1  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 22 – Листинг созданного каталога

Это было последнее задание II части лабораторной работы. Переходим к выполнению заданий III части лабораторной работы.

Часть III

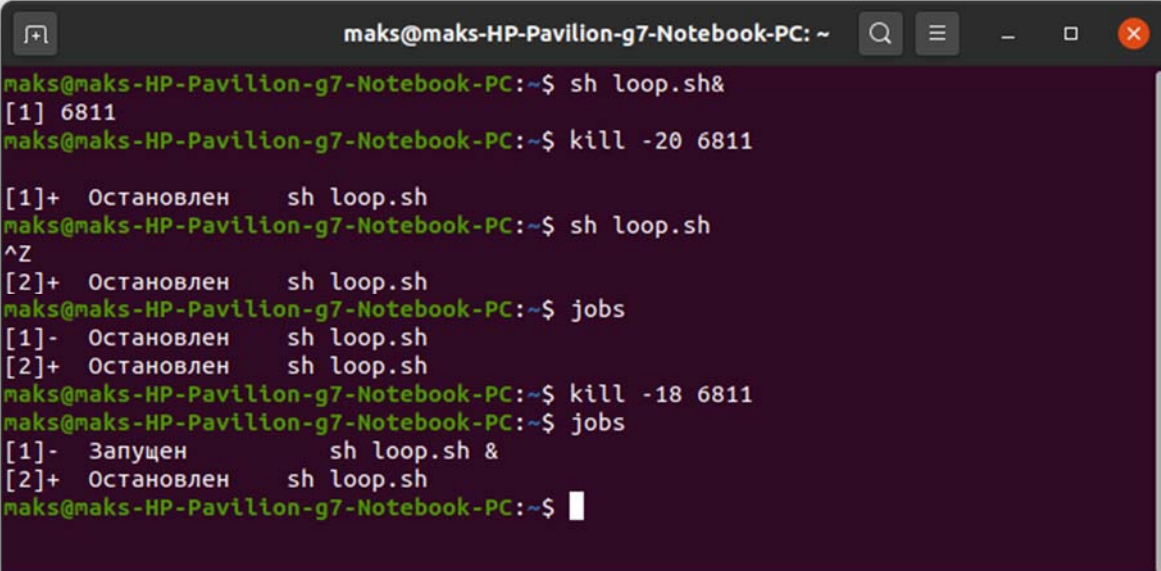
Для начала нам нужно сгенерировать следующую информацию — полный листинг в длинном формате о процессах текущего пользователя: PID, PPID, выделенное время ЦП, время запуска, размер образа. Сделаем это с помощью команды `ps`, но после аргумента «-o» укажем интересующие нас столбцы, а после аргумента «-u» — имя пользователя:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -o pid,ppid,c,stime,rss -u maks  
  PID  PPID  C  STIME  RSS  
  1023     1  0  10:28 10604  
  1024  1023  0  10:28  3584  
  1073  1023  0  10:28 18900  
  1075  1023  0  10:28 24744  
  1082     1  0  10:28  7464  
  1087   991  0  10:28  6620  
  1090  1087  3  10:28 76776  
  1096  1023  0  10:28  8272
```

Рисунок 23 – Процессы пользователя в нужном формате

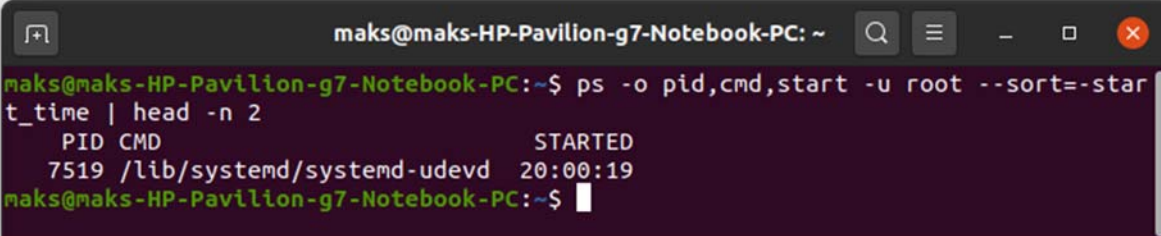
Затем нужно создать процесс и остановить его с помощью команды `kill -20` (сигнал SIGTSTP) и с помощью комбинации клавиш CTRL + Z:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh&  
[1] 6811  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ kill -20 6811  
  
[1]+  Остановлен    sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ sh loop.sh  
^Z  
[2]+  Остановлен    sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ jobs  
[1]-  Остановлен    sh loop.sh  
[2]+  Остановлен    sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ kill -18 6811  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ jobs  
[1]-  Запущен       sh loop.sh &  
[2]+  Остановлен    sh loop.sh  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 24 – Остановка процесса

Наконец, определим идентификатор и имя процесса, созданного последним пользователем root. Для этого выведем нужные нам поля, отсортируем таблицу по убыванию времени и возьмём шапку и первую строку этой таблицы:



```
maks@maks-HP-Pavilion-g7-Notebook-PC: ~  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$ ps -o pid,cmd,start -u root --sort=-start_time | head -n 2  
PID CMD STARTED  
7519 /lib/systemd/systemd-udevd 20:00:19  
maks@maks-HP-Pavilion-g7-Notebook-PC:~$
```

Рисунок 25 – Последний процесс, созданный пользователем root

Вывод

В ходе выполнения лабораторной работы я ознакомился на практике с понятием процесса в операционной системе и приобрёл опыт и навыки управления процессами в операционной системе Linux.