



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ДИСЦИПЛИНА «Операционные системы»

Лабораторная работа № 3

Тема «Знакомство с Shell»

Студентка Садулаева Т. Р.

Группа ИУ7-54Б

Оценка (баллы) _____

Преподаватель Рязанова Н.Ю.

Москва.
2020 г.

Задание 1. Изучение команд shell.

Используя команду `mkdir` создайте директорию именем своей группы. Например, `$ mkdir iu7`; перейдите в созданную директорию с помощью команды `cd`.

```
Teo@Teo code % mkdir iu7-54b
Teo@Teo code % cd iu7-54b
Teo@Teo iu7-54b %
```

Создайте поддиректорию, например, используя свою фамилию.

```
Teo@Teo iu7-54b % mkdir Sadulaeva
Teo@Teo iu7-54b % cd Sadulaeva
Teo@Teo Sadulaeva % pwd
/Users/teona/code/iu7-54b/Sadulaeva
```

Используя `man` изучите команды, список которых приведен ниже:

- Команда `ls`:

```
Teo@Teo Sadulaeva % man ls
```

```
LS(1)                                BSD General Commands Manual                                LS(1)

NAME
  ls -- list directory contents

SYNOPSIS
  ls [-ABCFGHLOPRSTUW@abcdefghiklmnopqrstuwx1] [file ...]

DESCRIPTION
  For each operand that names a file of a type other than directory, ls displays its
  name as well as any requested, associated information. For each operand that names a
  file of type directory, ls displays the names of files contained within that direc-
  tory, as well as any requested, associated information.

  If no operands are given, the contents of the current directory are displayed. If
  more than one operand is given, non-directory operands are displayed first; directory
  and non-directory operands are sorted separately and in lexicographical order.

  The following options are available:

  -@      Display extended attribute keys and sizes in long (-l) output.

  -1      (The numeric digit ``one''.) Force output to be one entry per line. This is
  the default when output is not to a terminal.
```

- Команда ps:

```
Teo@Teo Sadulaeva % man ps
```

```
PS(1) BSD General Commands Manual PS(1)

NAME
    ps -- process status

SYNOPSIS
    ps [-AaCcEefhjLMmrSTvwXx] [-O fmt | -o fmt] [-G gid[,gid...]] [-g grp[,grp...]]
      [-u uid[,uid...]] [-p pid[,pid...]] [-t tty[,tty...]] [-U user[,user...]]
    ps [-L]

DESCRIPTION
    The ps utility displays a header line, followed by lines containing information about
    all of your processes that have controlling terminals.

    A different set of processes can be selected for display by using any combination of
    the -a, -G, -g, -p, -T, -t, -U, and -u options. If more than one of these options
    are given, then ps will select all processes which are matched by at least one of the
    given options.

    For the processes which have been selected for display, ps will usually display one
    line per process. The -M option may result in multiple output lines (one line per
    thread) for some processes. By default all of these output lines are sorted first by
    controlling terminal, then by process ID. The -m, -r, and -v options will change the
    sort order. If more than one sorting option was given, then the selected processes
```

Используя команду man в описании команды ps, найдите флаги (flags) и статус (state).

```
PROCESS FLAGS
    The sum of these values is displayed in the "F" column, which is provided by the flags output specifier:

        1   forked but didn't exec
        4   used super-user privileges

PROCESS STATE CODES
    Here are the different values that the s, stat and state output specifiers (header "STAT" or "S") will display:

        D   uninterruptible sleep (usually IO)
        I   Idle kernel thread
        R   running or runnable (on run queue)
        S   interruptible sleep (waiting for an event to complete)
        T   stopped by job control signal
        t   stopped by debugger during the tracing
        W   paging (not valid since the 2.6.xx kernel)
        X   dead (should never be seen)
        Z   defunct ("zombie") process, terminated but not reaped by its parent

    For BSD formats and when the stat keyword is used, additional characters may be displayed:

        <   high-priority (not nice to other users)
        N   low-priority (nice to other users)
        L   has pages locked into memory (for real-time and custom IO)
        s   is a session leader
        l   is multi-threaded (using CLONE_THREAD, like NPTL pthreads do)
        +   is in the foreground process group

OBSOLETE SORT KEYS
```

Задание 2.

Напишите программу, в которой создается дочерний процесс и организуйте как в предке, так и в потомке бесконечные циклы, в которых выводятся идентификаторы процессов с помощью системного вызова getpid().

Листинг 1. – Код программы задания 2.

```
#include <stdio.h>
#include <unistd.h>

int main()
{
    int childpid;

    if (childpid = fork() == -1)
    {
        perror("Can't fork.\n");
        return 1;
    }
    else if (childpid == 0)
    {
        while (1) printf(" %d", getpid());
        return 0;
    }
    else
    {
        while (1) printf("%d ", getpid());
        return 0;
    }
}
```

1. Запустите программу и посмотрите идентификаторы созданных процессов: предка и потомка;

Teo@Teo Sadulaeva % ps -al

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
4	S	1000	1668	1665	2	80	0	-	206294	ep_pol	tty2	00:00:02	Xorg
0	S	1000	1698	1665	0	80	0	-	47762	poll_s	tty2	00:00:00	gnome-sess
0	S	1000	3395	3250	50	80	0	-	622	tty_wr	pts/0	00:00:02	a.out
1	S	1000	3396	3395	50	80	0	-	622	wait_w	pts/0	00:00:02	a.out
0	R	1000	3405	3399	0	80	0	-	2861	-	pts/1	00:00:00	ps

Созданные нами процессы:

0	S	1000	1698	1665	0	80	0	-	47762	poll_s	tty2	00:00:00	gnome-sess
0	S	1000	3395	3250	50	80	0	-	622	tty_wr	pts/0	00:00:02	a.out
1	S	1000	3396	3395	50	80	0	-	622	wait_w	pts/0	00:00:02	a.out
0	R	1000	3405	3399	0	80	0	-	2861	-	pts/1	00:00:00	ps

2. Для получения процесса зомби выполните следующие действия: а) удалите командой `kill` потомка и посмотрите с помощью команды `ps` его новый статус – Z; б) удалите предка.

```
Teo@Teo Sadulaeva % kill 3396
```

```
Teo@Teo Sadulaeva % ps -al
```

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
4	S	1000	1668	1665	2	80	0	-	206352	ep_pol	tty2	00:00:04	Xorg
0	S	1000	1698	1665	0	80	0	-	47762	poll_s	tty2	00:00:00	gnome-sess
0	R	1000	3395	3250	53	80	0	-	622	-	pts/0	00:00:25	a.out
1	Z	1000	3396	3395	50	80	0	-	0	-	pts/0	00:00:23	
0	R	1000	3416	3399	0	80	0	-	2861	-	pts/1	00:00:00	ps

3. Для получения «осиротевшего» процесса запустите программу еще раз, но в этот раз удалите предка и посмотрите с помощью команды `ps` идентификатор предка у продолжающего выполняться потомка – идентификатор предка будет изменен на 1, так как процесс был «усыновлен» процессом с идентификатором 1 процессом «открывшим» терминал в случае, если используется Unix BSD, или идентификатор процессов-посредников в случае, Linux Ubuntu.

Созданные нами процессы:

0	S	1000	1698	1665	0	80	0	-	47762	poll_s	tty2	00:00:00	gnome-sess
0	S	1000	3493	3250	43	80	0	-	622	wait_w	pts/0	00:00:01	a.out
1	S	1000	3494	3493	43	80	0	-	622	tty_wr	pts/0	00:00:01	a.out
0	R	1000	3495	3399	0	80	0	-	2861	-	pts/1	00:00:00	ps

Удаляем предка:

```
Teo@Teo Sadulaeva % kill 3394
```

Видим, что осиротевший процесс был усыновлен процессом 1571.

0	S	1000	1698	1665	0	80	0	-	47762	poll_s	tty2	00:00:00	gnome-sess
1	R	1000	3494	1571	52	80	0	-	622	-	pts/0	00:00:26	a.out
0	R	1000	3520	3399	0	80	0	-	2861	-	pts/1	00:00:00	ps

Задание 3.

Создадим жесткую и символическую ссылку на файл.

```
[Teo@Teo Sadulaeva % ln example_file.txt hardlink
```

```
[Teo@Teo Sadulaeva % ln example.txt softling -s
```

```
[Teo@Teo Sadulaeva % ls -al
```

```
[Teo@Teo Sadulaeva % ls -al
total 0
drwxr-xr-x  5 Teo  staff  160 Dec 28 21:55 .
drwxr-xr-x  3 Teo  staff   96 Dec 28 21:41 ..
-rw-r--r--  3 Teo  staff    0 Dec 28 21:52 example.txt
-rw-r--r--  3 Teo  staff    0 Dec 28 21:52 hardlink
-rw-r--r--  3 Teo  staff    0 Dec 28 21:52 softling
```

Создадим именованный программный канал командой mknod.

```
[Teo@Teo Sadulaeva % mknod pipe p
```

Изменим консоль и с помощью команды tee получим через канал текст, выведем его на экран.

```
Teo@Teo Sadulaeva % echo echo example > pipe
Teo@Teo Sadulaeva % tee < pipe
echo example
```

Задание 4.

Написать программу вывода на экран сообщения «Hello, World!»

Листинг 2. – Код программы задания 4.

```
#include <stdio>

int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```



```
Teo@Teo Sadulaeva % g++ task.cpp -o main
```

```
Teo@Teo Sadulaeva % ./main
```

```
Hello, World!
```

Задание 5. Вывод inode на экран

С помощью команд `ls -i` или `ls -li`.

```
Teo@Teo sadulaeva % ls -i
30521072 example.txt    30521072 hardlink      30521279 pipe           30521072 softling
Teo@Teo sadulaeva % ls -li
total 8
30521072 -rw-r--r--  3 Teo  staff   0 Dec 28 21:52 example.txt
30521072 -rw-r--r--  3 Teo  staff   0 Dec 28 21:52 hardlink
30521279 -rw-r--r--  1 Teo  staff  13 Dec 28 21:57 pipe
30521072 -rw-r--r--  3 Teo  staff   0 Dec 28 21:52 softling
```

Задание 6. Изменение приоритета процесса

С помощью команды `nice`. Создаем два процесса с PID 37052, 37051 с помощью программы из задания №2. Изменяем значение `nice` 37052 на 15.

Приоритет понижен с 31 до 16.

```
Teo@Teo sadulaeva % ps -al
  UID    PID  PPID      F  CPU  PRI  NI      SZ    RSS  WCHAN    S
    0  36908  1180    4106    0   31   0  4341772   3644  -      Ss
   501  36909  36908    4006    0   31   0  4325916   1728  -      S
   501  37051  36909    4006    0   31   0  4277424    728  -      S+
   501  37052  37051        6    0   31   0  4277424    384  -      R+
    0  36996  1180    4106    0   31   0  4349964   3652  -      Ss
   501  36997  36996    4006    0   31   0  4325916   1716  -      S
    0  37053  36997    4106    0   31   0  4277116   1060  -      R+
Teo@Teo sadulaeva % renice -n 15 -p 37052
Teo@Teo sadulaeva % ps -al
  UID    PID  PPID      F  CPU  PRI  NI      SZ    RSS  WCHAN    S
    0  36908  1180    4106    0   31   0  4341772   3644  -      Ss
   501  36909  36908    4006    0   31   0  4325916   1728  -      S
   501  37051  36909    4006    0   31   0  4277424    728  -      R+
   501  37052  37051        6    0  16  15  4277424    384  -      RN+
    0  36996  1180    4106    0   31   0  4349964   3652  -      Ss
   501  36997  36996    4006    0   31   0  4325916   1772  -      S
    0  37055  36997    4106    0   31   0  4279164   1080  -      R+
```