

Отчёт по лабораторной работе 6

Управление процессами

Вишняков Родион Сергеевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	17
4	Контрольные вопросы	18

Список иллюстраций

2.1	root	6
2.2	Ввод команд	6
2.3	Jobs	6
2.4	Фоновый режим	7
2.5	Перемещение заданий	7
2.6	Ввод команд	7
2.7	Тор	8
2.8	Задание dd	8
2.9	Ввод команд	9
2.10	Просмотр строк с dd	9
2.11	Замена приоритета	9
2.12	Просмотр строк	10
2.13	Остановка родительского процесса	10
2.14	Самостоятельная работа	10
2.15	Самостоятельная работа	11
2.16	Самостоятельная работа	11
2.17	Самостоятельная работа	12
2.18	Самостоятельная работа	12
2.19	Самостоятельная работа	13
2.20	Самостоятельная работа	13
2.21	Самостоятельная работа	13
2.22	Самостоятельная работа	14
2.23	Самостоятельная работа	14
2.24	Самостоятельная работа	14
2.25	Самостоятельная работа	15
2.26	Самостоятельная работа	15
2.27	Самостоятельная работа	16

Список таблиц

1 Цель работы

Получить навыки управления процессами операционной системы.

2 Выполнение лабораторной работы

Получаем полномочия администратора

```
rodvish@rodvishh:~$ sudo -i
[sudo] пароль для rodvish:
root@rodvishh:~#
```

Рис. 2.1: root

Вводим предоставленные команды

```
root@rodvishh:~# sleep 3600 &
[1] 4918
root@rodvishh:~# dd if=/dev/zero of=/dev/null &
[2] 4959
root@rodvishh:~# sleep 7200
^Z
[3]+  Остановлен    sleep 7200
root@rodvishh:~#
```

Рис. 2.2: Ввод команд

Вводим команду jobs

```
root@rodvishh:~# jobs
[1]  Запущен          sleep 3600 &
[2]- Запущен          dd if=/dev/zero of=/dev/null &
[3]+ Остановлен      sleep 7200
root@rodvishh:~#
```

Рис. 2.3: Jobs

Вводим команду для продолжения выполнения задания 3 в фоновом и смотрим изменения

```

root@rodvishh:~# bg 3
[3]+ sleep 7200 &
root@rodvishh:~# jobs
[1]  Запущен          sleep 3600 &
[2]-  Запущен          dd if=/dev/zero of=/dev/null &
[3]+  Запущен          sleep 7200 &
root@rodvishh:~# █

```

Рис. 2.4: Фоновый режим

Перемещаем задания 1,2,3 на передний план

```

root@rodvishh:~# fg 1
sleep 3600
^C
root@rodvishh:~# jobs
[2]-  Запущен          dd if=/dev/zero of=/dev/null &
[3]+  Запущен          sleep 7200 &
root@rodvishh:~# fg 2
dd if=/dev/zero of=/dev/null
^C307990938+0 records in
307990937+0 records out
157691359744 bytes (158 GB, 147 GiB) copied, 224,921 s, 701 MB/s

root@rodvishh:~# jobs
[3]+  Запущен          sleep 7200 &
root@rodvishh:~# fg 3
sleep 7200
^C
root@rodvishh:~# jobs
root@rodvishh:~# █

```

Рис. 2.5: Перемещение заданий

Вводим предоставленные команды

```

rodvish@rodvishh:~$ dd if=/dev/zero of=/dev/null &
[1] 6167
rodvish@rodvishh:~$ exit

```

Рис. 2.6: Ввод команд

Запускаем top

```

top - 11:58:06 up 11 min, 4 users, load average: 1,80, 1,23, 0,76
Tasks: 246 total, 4 running, 242 sleeping, 0 stopped, 0 zombie
%Cpu(s): 28,7 us, 70,0 sy, 0,5 ni, 0,0 id, 0,0 wa, 0,7 hi, 0,0 si, 0,0 st
MiB Mem : 3653,5 total, 260,2 free, 2434,5 used, 1257,6 buff/cache
MiB Swap: 4040,0 total, 4034,2 free, 5,8 used, 1219,1 avail Mem

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6167	rodvish	20	0	227308	1996	1996	R	85,4	0,1	0:19.65	dd
5995	rodvish	20	0	227308	2056	2056	R	74,8	0,1	1:16.83	dd
2426	rodvish	20	0	4260788	517900	145740	S	23,3	13,8	1:27.44	gnome-shell
4676	rodvish	20	0	1946076	335384	98736	R	6,0	9,0	0:06.47	ptxvis
11	root	20	0	0	0	0	I	2,3	0,0	0:01.05	kworker/u8:0-events_unbound
38	root	20	0	0	0	0	S	0,7	0,0	0:00.06	kcompactd0
27	root	20	0	0	0	0	S	0,3	0,0	0:00.07	ksoftirqd/1
4487	rodvish	20	0	2753784	151008	97492	S	0,3	4,0	0:01.68	Isolated Web Co
5138	rodvish	20	0	1956304	265488	108524	S	0,3	7,1	0:05.89	nautilus
6308	rodvish	20	0	231756	5596	3420	R	0,3	0,1	0:00.02	top
1	root	20	0	49196	41360	10364	S	0,0	1,1	0:03.35	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0,0	0,0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-rcu_gp
5	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-sync_wq
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-slab_flushwq
7	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-netns
9	root	20	0	0	0	0	I	0,0	0,0	0:00.12	kworker/0:1-events_power_eff
10	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/0:0H-events_highpri
12	root	20	0	0	0	0	I	0,0	0,0	0:01.65	kworker/u8:1-flush-253:0
13	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-mm_percpu_wq
14	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_kthread
15	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_rude_kthread
16	root	20	0	0	0	0	I	0,0	0,0	0:00.00	rcu_tasks_trace_kthread
17	root	20	0	0	0	0	S	0,0	0,0	0:00.07	ksoftirqd/0
18	root	20	0	0	0	0	I	0,0	0,0	0:00.41	rcu_preempt
19	root	20	0	0	0	0	S	0,0	0,0	0:00.00	rcu_exp_par_gp_kthread_worke
20	root	20	0	0	0	0	S	0,0	0,0	0:00.07	rcu_exp_gp_kthread_worker
21	root	rt	0	0	0	0	S	0,0	0,0	0:00.01	migration/0
22	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_inject/0
23	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/0
24	root	20	0	0	0	0	S	0,0	0,0	0:00.00	cpuhp/1
25	root	-51	0	0	0	0	S	0,0	0,0	0:00.00	idle_inject/1
26	root	rt	0	0	0	0	S	0,0	0,0	0:00.65	migration/1
29	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/1:0H-events_highpri
31	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kdevtmpfs

Рис. 2.7: Тор

Вновь используем top и убиваем задание dd

```

top - 11:59:14 up 12 min, 4 users, load average: 2,45, 1,56, 0,91
Tasks: 242 total, 3 running, 239 sleeping, 0 stopped, 0 zombie
%Cpu(s): 21,7 us, 78,3 sy, 0,0 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 3653,5 total, 201,4 free, 2492,0 used, 1256,0 buff/cache
MiB Swap: 4040,0 total, 4034,2 free, 5,8 used, 1161,5 avail Mem

```

PID to signal/kill [default pid = 6167]

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6167	rodvish	20	0	227308	1996	1996	R	100,0	0,1	1:17.41	dd
5995	rodvish	20	0	227308	2056	2056	R	81,8	0,1	2:14.62	dd
6527	rodvish	20	0	231616	5060	3012	R	9,1	0,1	0:00.01	top
1	root	20	0	49196	41360	10364	S	0,0	1,1	0:03.46	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0,0	0,0	0:00.00	pool_workqu
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-r
5	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-s
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-s
7	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-n
9	root	20	0	0	0	0	I	0,0	0,0	0:00.13	kworker/0:1

Рис. 2.8: Задание dd

Вводим предоставленные команды


```

root@rodvishh:~# dd if=/dev/zero of=/dev/null &
[1] 6644
root@rodvishh:~# dd if=/dev/zero of=/dev/null &
[2] 6684
root@rodvishh:~# dd if=/dev/zero of=/dev/null &
[3] 6687
root@rodvishh:~#

```

Рис. 2.9: Ввод команд

Смотрим все строки, в которых есть буквы dd

```

root@rodvishh:~# ps aux | grep dd
root      2  0.0  0.0      0   0 ?        S   11:46   0:00 [kthreaddd]
root     71  0.0  0.0      0   0 ?        I<  11:46   0:00 [kworker/R-ipv6_addrconf]
root    1104  0.0  0.0 578492 3296 ?        Sl  11:46   0:00 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
rodvishh 2834  0.0  0.6 1037868 24232 ?        Ssl  11:47   0:00 /usr/libexec/evolution-addrbook-factory
rodvishh 3490  0.0  0.9 243780 36912 ?        Sl  11:47   0:00 /usr/lib64/firefox/firefox -contentproc -parentBuildID 20250826043737 -sandboxin
gKind 0 -prefsLen 34414 -prefMapSize 249431 -appDir /usr/lib64/firefox/browser [8f406dd6-041d-4151-a880-b4b7bfd29194] 3290 utility
rodvishh 3614  0.0  0.9 241724 35872 ?        Sl  11:47   0:00 /usr/lib64/firefox/firefox -contentproc -parentBuildID 20250826043737 -prefsLen
34521 -prefMapSize 249431 -appDir /usr/lib64/firefox/browser [153b92e9-da73-4bd2-bfef-543a274c75a6] 3290 rdd
rodvishh 4527  0.0  1.9 2632296 71412 ?        Sl  11:50   0:00 /usr/lib64/firefox/firefox -contentproc -childID 25 -isForBrowser -prefsLen 3201
4 -prefMapSize 249431 -jsInitLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.
ja -appDir /usr/lib64/firefox/browser [41a01435-c9c3-4a6f-bdfe-12a04243416c] 3290 tab
rodvishh 5204  0.0  0.2 683352 9848 ?        Sl  11:53   0:00 /usr/libexec/gvfsd-msdd --spawner /1.16 /org/gtk/gvfs/exec_spaw/4
rodvishh 5210  0.0  0.7 259072 28952 ?        S   11:53   0:00 /usr/bin/python3 /usr/bin/msdd --no-host --discovery --listen /run/user/1000/gvf
sd/msdd
rodvishh 5995 87.9  0.0 227388 2056 ?        R   11:56   3:37 dd if=/dev/zero of=/dev/null
root     6644 52.5  0.0 227388 2156 pts/1    R   11:59   0:28 dd if=/dev/zero of=/dev/null
root     6684 33.9  0.0 227388 2088 pts/1    R   12:00   0:12 dd if=/dev/zero of=/dev/null
root     6687 32.7  0.0 227388 2116 pts/1    R   12:00   0:11 dd if=/dev/zero of=/dev/null
root     6782  0.0  0.0 227712 2224 pts/1    S+  12:00   0:00 grep --color=auto dd
root@rodvishh:~#

```

Рис. 2.10: Просмотр строк с dd

Меняем приоритет

```

root@rodvishh:~# renice -n 5 6684
6684 (process ID) old priority 0, new priority 5
root@rodvishh:~#

```

Рис. 2.11: Замена приоритета

Смотрим строки с параметров -B5

```

root@rodvishh:~# ps fax | grep -B 5 dd
PID TTY          STAT       TIME COMMAND
  2 ?            S          0:00 [kthreadd]

...

61 ?          I<         0:00 \_ [kworker/R-kthrotld]
66 ?          I<         0:00 \_ [kworker/R-acpi_thermal_pm]
67 ?          I<         0:00 \_ [kworker/R-kmpath_rdacd]
68 ?          I<         0:00 \_ [kworker/R-kmsad]
70 ?          I<         0:00 \_ [kworker/R-mlb]
71 ?          I<         0:00 \_ [kworker/R-ipv6_addrconf]

...

873 ?         Sns        0:00 /usr/sbin/alsactl -s -n 19 -c -E ALSA_CONFIG_PATH=/etc/alsa/alsactl.conf --initfile=/lib/alsa/init/00main.rdaemon
877 ?         S          0:00 /usr/sbin/chronyd -F 2
909 ?         Ssl        0:00 /usr/sbin/ModemManager
910 ?         Ssl        0:00 /usr/sbin/pttend -sP /usr/sbin/firewalld --nofork --nopid
1102 ?        Sl         0:00 /usr/sbin/VBoxService --pidfile /var/run/vboxadd.service.sh
1104 ?        Sl         0:00 /usr/sbin/VBoxService --pidfile /var/run/vboxadd.service.sh

...

3081 ?        Sl         0:00 | \_ /usr/libexec/mutter-x11-frames
3290 ?        Sl         1:21 | \_ /usr/lib64/firefox/firefox
3362 ?        Sl         0:00 | \_ /usr/lib64/firefox/firefox -contentproc -parentBuildID 20250826043737 -prefsLen 28522 -prefMapSize 249431
-appDir /usr/lib64/firefox/browser [libb0b378-5c7f-42a1-a45c-796589f0d9d2] 3290 socket
3393 ?        Sl         0:00 | \_ /usr/lib64/firefox/firefox -contentproc -childID 1 -isForBrowser -prefsLen 28587 -prefMapSize 249431 -jsI
tLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.ja -appDir /usr/lib64/fire
ox/browser [d21d7720-f2b7-4bfd-b940-260e6ade2dc9] 3290 tab
3425 ?        Sl         0:01 | \_ /usr/lib64/firefox/firefox -contentproc -childID 2 -isForBrowser -prefsLen 29547 -prefMapSize 249431 -jsI
tLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.ja -appDir /usr/lib64/fire
ox/browser [3ea3d58-a401-4d44-948c-8438634556f9] 3290 tab
3499 ?        Sl         0:00 | \_ /usr/lib64/firefox/firefox -contentproc -parentBuildID 20250826043737 -sandboxingKind 0 -prefsLen 34414 -
refMapSize 249431 -appDir /usr/lib64/firefox/browser [8f40046-041d-4151-a880-b4b7bfd29194] 3290 utility
3561 ?        Sl         0:23 | \_ /usr/lib64/firefox/firefox -contentproc -childID 5 -isForBrowser -prefsLen 31763 -prefMapSize 249431 -jsI
tLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.ja -appDir /usr/lib64/fire
ox/browser [7efb48c5-6939-475e-b7cc-9d637db4222a] 3290 tab
3614 ?        Sl         0:00 | \_ /usr/lib64/firefox/firefox -contentproc -parentBuildID 20250826043737 -prefsLen 34521 -prefMapSize 249431
-appDir /usr/lib64/firefox/browser [153b92e9-da73-4bd2-bfef-543a274c75a0] 3290 rd
4487 ?        Sl         0:05 | \_ /usr/lib64/firefox/firefox -contentproc -childID 22 -isForBrowser -prefsLen 32014 -prefMapSize 249431 -js
ntLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.ja -appDir /usr/lib64/fire
fox/browser [42d37980-a9ce-4627-aba9-181cbad10f18] 3290 tab
4510 ?        Sl         0:00 | \_ /usr/lib64/firefox/firefox -contentproc -childID 24 -isForBrowser -prefsLen 32014 -prefMapSize 249431 -js
ntLen 234912 -parentBuildID 20250826043737 -greomni /usr/lib64/firefox/omni.ja -appomni /usr/lib64/firefox/browser/omni.ja -appDir /usr/lib64/fire
fox/browser [8aa50950-1f80-4514-86df-d06d668a736b] 3290 tab

```

Рис. 2.12: Просмотр строк

Останавливаем родительский процесс

```

root@rodvishh:~# kill -9 4832
Убито
rodvishh@rodvishh:~$

```

Рис. 2.13: Остановка родительского процесса

Выполняем задание для самостоятельной работы

```

rodvishh@rodvishh:~$ dd if=/dev/zero of=/dev/null &
[2] 8191
rodvishh@rodvishh:~$ dd if=/dev/zero of=/dev/null &
[3] 8193
rodvishh@rodvishh:~$ dd if=/dev/zero of=/dev/null &
[4] 8205

```

Рис. 2.14: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

root@rodvishh:~# renice -n -15 8191
8191 (process ID) old priority -5, new priority -15
root@rodvishh:~# kill -9 8191
root@rodvishh:~# kill -9 8193
root@rodvishh:~# kill -9 8205
root@rodvishh:~#

```

Рис. 2.15: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

root@rodvishh:~# yes > /dev/null &
[4] 54530
root@rodvishh:~# yes > /dev/null
^Z
[5]+  Остановлен    yes > /dev/null
root@rodvishh:~# jobs
[2]  Остановлен    yes > /dev/null
[3]-  Остановлен    yes
[4]  Запущен       yes > /dev/null &
[5]+  Остановлен    yes > /dev/null
root@rodvishh:~# fg %1

```

Рис. 2.16: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

y
y
y
y^Z
[6]+  Остановлен      yes
root@rodvishh:~# jobs
[2]   Остановлен      yes > /dev/null
[3]   Остановлен      yes
[4]   Запущен         yes > /dev/nu
[5]-  Остановлен      yes > /dev/null
[6]+  Остановлен      yes
root@rodvishh:~# fg %2
yes > /dev/null
^C
root@rodvishh:~# █

```

Рис. 2.17: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

root@rodvishh:~# jobs
[4]   Запущен         yes > /dev/null &
[5]-  Остановлен      yes > /dev/null
[6]+  Остановлен      yes
root@rodvishh:~# █

```

Рис. 2.18: Самостоятельная работа

Выполняем задание для самостоятельной работы

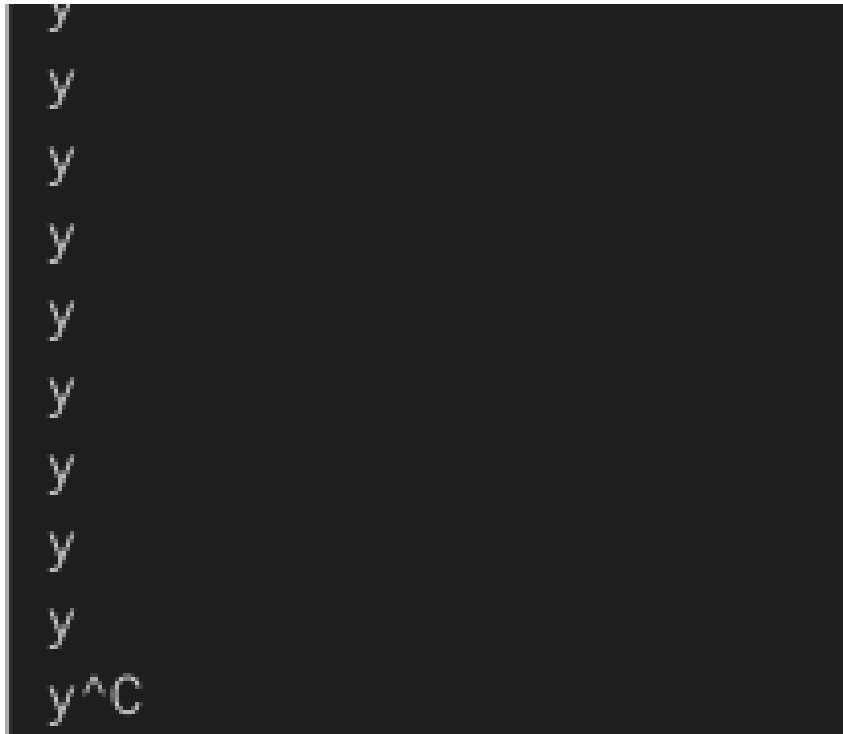


Рис. 2.19: Самостоятельная работа

Выполняем задание для самостоятельной работы

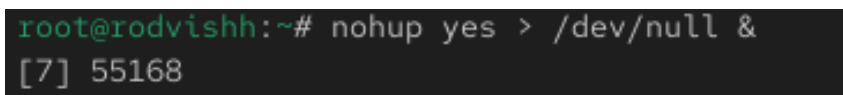


Рис. 2.20: Самостоятельная работа

Выполняем задание для самостоятельной работы

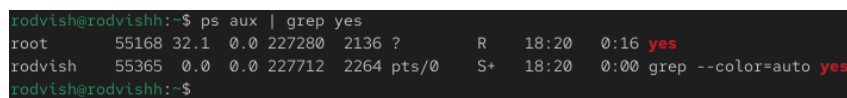


Рис. 2.21: Самостоятельная работа

Выполняем задание для самостоятельной работы

```
top - 18:21:14 up 6:34, 5 users, load average: 6,39, 5,60, 4,87
Tasks: 246 total, 6 running, 240 sleeping, 0 stopped, 0 zombie
%Cpu(s): 26,7 us, 73,3 sy, 0,0 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 3653,5 total, 346,6 free, 2779,3 used, 900,5 buff/cache
MiB Swap: 4040,0 total, 4018,4 free, 21,6 used, 874,2 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
5995	rodvish	20	0	227308	2056	2056	R	83,3	0,1	354:45.45	dd
55168	root	20	0	227280	2136	2136	R	50,0	0,1	0:25.94	yes
6687	root	20	0	227308	2116	2116	R	33,3	0,1	221:23.75	dd
6684	root	25	5	227308	2008	2008	R	8,3	0,1	75:22.81	dd
55413	rodvish	20	0	231608	5060	3012	R	8,3	0,1	0:00.01	top
1	root	20	0	49196	41360	10364	S	0,0	1,1	0:44.92	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0,0	0,0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-rcu_gp
5	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-sync_wq
6	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-slab_flushwq
7	root	0	-20	0	0	0	I	0,0	0,0	0:00.00	kworker/R-netns

Рис. 2.22: Самостоятельная работа

Выполняем задание для самостоятельной работы

```
rodvish@rodvishh:~$ yes > /dev/null &
[5] 55610
rodvish@rodvishh:~$ yes > /dev/null &
[6] 55612
rodvish@rodvishh:~$ yes > /dev/null &
[7] 55621
rodvish@rodvishh:~$ kill 55610
[5] Завершено yes > /dev/null
rodvish@rodvishh:~$ kill %6
[6] Завершено yes > /dev/null
rodvish@rodvishh:~$
```

Рис. 2.23: Самостоятельная работа

Выполняем задание для самостоятельной работы

```
rodvish@rodvishh:~$ kill -1 55621
[7]- Обрыв терминальной линии yes > /dev/null
rodvish@rodvishh:~$ nohup yes > /dev/null &
[5] 55963
nohup: rodvish@rodvishh:~$ ввод игнорируется и поток ошибок перенаправляется на стандартный вывод
nohup yes > /dev/nul
rodvish@rodvishh:~$ kill -1 55963
rodvish@rodvishh:~$
```

Рис. 2.24: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

rodvish@rodvishh:~$ yes > /dev/null &
[6] 56092
rodvish@rodvishh:~$ yes > /dev/null &
[7] 56097
rodvish@rodvishh:~$ yes > /dev/null &
[8] 56099
rodvish@rodvishh:~$ killall yes
yes(55168): Операция не позволена
[5]   Завершено      nohup yes > /dev/null
[6]   Завершено      yes > /dev/null
[7]   Завершено      yes > /dev/null
[8]-  Завершено      yes > /dev/null
rodvish@rodvishh:~$

```

Рис. 2.25: Самостоятельная работа

Выполняем задание для самостоятельной работы

```

rodvish@rodvishh:~$ yes > /dev/null &
[5] 56232
[4]   Завершено      yes > /dev/null
rodvish@rodvishh:~$ nice -n -5 yes > /dev/null &
[6] 56273
rodvish@rodvishh:~$ nice: невозможно установить значение nice: Отказано в доступе

rodvish@rodvishh:~$ ps -o pid,ni.cmd -C yes
error: unknown user-defined format specifier "ni.cmd"

Usage:
ps [options]

Try 'ps --help <simple|list|output|threads|misc|all>'
or 'ps --help <s|l|o|t|m|a>'
for additional help text.

For more details see ps(1).
rodvish@rodvishh:~$

```

Рис. 2.26: Самостоятельная работа

Выполняем задание для самостоятельной работы

```
rodvish@rodvishh:~$ ps -C yes
  PID TTY          TIME CMD
 55168 ?            00:03:41 yes
 56232 pts/0        00:00:28 yes
 56273 pts/0        00:00:18 yes
rodvish@rodvishh:~$ sudo renice -n -5 56232
[sudo] пароль для rodvish:
56232 (process ID) old priority 0, new priority -5
rodvish@rodvishh:~$ ps -o pid,ni,cmd -C yes
  PID  NI CMD
 55168   0 yes
 56232  -5 yes
 56273   0 yes
rodvish@rodvishh:~$
```

Рис. 2.27: Самостоятельная работа

3 Вывод

Мы получили навыки управления процессами операционной системы

4 Контрольные вопросы

1. Какая команда даёт обзор всех текущих заданий оболочки? Ответ: `jobs`
2. Как остановить текущее задание оболочки, чтобы продолжить его выполнение в фоновом режиме? Ответ:
 - Нажать `Ctrl+Z` для остановки (приостановки)
 - Затем выполнить `bg` для продолжения в фоновом режиме
3. Какую комбинацию клавиш можно использовать для отмены текущего задания оболочки? Ответ: `Ctrl+C`
4. Необходимо отменить одно из начатых заданий. Доступ к оболочке, в которой в данный момент работает пользователь, невозможен. Что можно сделать, чтобы отменить задание? Ответ:
 - Найти PID процесса: `ps aux | grep имя_процесса`
 - Завершить процесс: `kill PID`
 - Или завершить по имени: `killall имя_процесса`
5. Какая команда используется для отображения отношений между родительскими и дочерними процессами? Ответ: `ps fax` или `pstree`
6. Какая команда позволит изменить приоритет процесса с идентификатором 1234 на более высокий? Ответ: `renice -n -5 1234` (только для root) Для обычного пользователя можно только понизить приоритет: `renice -n 5 1234`
7. В системе в настоящее время запущено 20 процессов `dd`. Как проще всего остановить их все сразу? Ответ: `killall dd`
8. Какая команда позволяет остановить команду с именем `mycommand`? Ответ: `pkill mycommand` или `killall mycommand`

9. Какая команда используется в `top`, чтобы убить процесс? Ответ:

- В `top` нажать `k`
- Ввести PID процесса
- Ввести сигнал (по умолчанию 15, для принудительного завершения - 9)

10. Как запустить команду с достаточно высоким приоритетом, не рискуя, что не хватит ресурсов для других процессов? Ответ:

- Для обычного пользователя: `nice -n 10` команда (устанавливает низкий приоритет)
- Для `root`: можно установить отрицательное значение, но с осторожностью
- Альтернативно: использовать `ionice` для управления I/O приоритетом