

МІНІСТЕРСТВО ОСВІТИ Й НАУКИ УКРАЇНИ  
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ  
Інститут комп'ютерних систем  
Кафедра інформаційних систем

Лабораторна робота № 7  
За дисципліною: "Операційні системи"  
**Тема: «Команди управління процесами в ОС Unix»**

Виконав:  
Студентка групи AI-205  
Светашов Д.В.  
Перевірили:  
Блажко О.А.  
Дрозд М.О.

Одеса 2021

### Мета роботи:

отримання навичок в управлінні процесами в ОС Unix засобами командної оболонки.

Процес це програма на стадії виконання.

Нехай **оперативна пам'ять** на комп'ютері сервері становить **N Гб**. Системні процеси

**ОС** займають до **М Гбайт** пам'яті, а **кожна програма** користувача може використовувати до **К Гбайт** пам'яті . Нехай в середньому процеси програм користувачів витрачають F % свого часу на очікування завершення вводу виводу . Визначте середню завантаженість процесора , використовуючи значення з таблиці 3 варіантів

№ команди	№ учасника команди	Пам'ять, N Гб	ОС, М Гб	Програма, К Гб	Очікування завершення вводу/виводу, F
8	2	12	1	0.5	35

**Відповідь:** Розмір пам'яті який є доступним для розміщення програм дорівнює 11 Гб , так як система займає 1 Гб. Завдяки 11 Гб ми можемо розмістити 11/0.5 процесів ( 22 процеси). Час очікування для процесу - 35% знаходження в оперативній пам'яті. Тому середня завантаженість процесора дорівнює  $1 - 0.35^{22}$  , що дорівнює близько 100 %.

### Завдання

#### 2 Перегляд таблиці процесів

1. Отримайте ієрархію всіх процесів із зазначенням імен користувачів

```
[svetashov_danilo@vpsj3IeQ ~]$ pstree -u
systemd--NetworkManager--2*[{NetworkManager}]
--agetty
--auditd--{auditd}
--3*[bash(sherbakov_artur)]
--belobrov.sh(belobrov_artur)--belobrov.sh
--belobrov2.sh(belobrov_artur)--belobrov2.sh
--belobrov3.sh(belobrov_artur)--belobrov3.sh
--chronyd(chrony)
--crond
--dbus-daemon(dbus)
--exim(exim)
--gssproxy--5*[{gssproxy}]
--httpd--10*[httpd(apache)]
--irqbalance
--mysqld(mysql)--30*[{mysqld}]
--named(named)--4*[{named}]
--nano(bogachik_egor)
--nano(baranyuk_dmitro)
--nesterenko.sh(nesterenko_mikola)--nesterenko.sh
--nesterenko2.sh(nesterenko_mikola)--nesterenko2.sh
--nesterenko3.sh(nesterenko_mikola)--nesterenko3.sh
--nginx--nginx(emps)
--php-fpm--6*[php-fpm(soft)]
--php-fpm--3*[php-fpm]
--php-fpm--3*[php-fpm(emps)]
--php-fpm--3*[php-fpm(soft)]
--ping(oracle)
--ping(kelembet_evgen)
--ping(malofeev_denis)
--ping(stepanenko_gleb)
--ping(lobko_daniil)
--ping(kovach_dmitro)
--6*[ping(markovskij_danilo)]
--4*[ping(melnichenko_egor)]
```

2. Отримайте ієрархію процесів, запущених від імені вашого профілю і з зазначенням PID цих процесів.

```
[svetashov_danilo@vpsj3IeQ ~]$ pstree -p svetashov_danilo
sshd(657)--bash(673)--less(22849)
--pstree(12668)
[svetashov_danilo@vpsj3IeQ ~]$
```

3. Отримайте список процесів, запущених в поточному терміналі, зі спрощеним набором колонок виведення таблиці процесів

```
[svetashov_danilo@vpsj3IeQ ~]$ ps -o pid,ppid,cmd
PID  PPID  CMD
673   657   -bash
9566  673   ps -o pid,ppid,cmd
22849 673   less
```

4. Отримайте список процесів, запущених від імені вашого користувача, з розширеним набором колонок таблиці процесів

```
[svetashov_danilo@vpsj3IeQ ~]$ pstree svetashov_danilo | ps -F
UID      PID  PPID  C   SZ   RSS  PSR  STIME  TTY      TIME  CMD
svetash+  673   657   0 28920 2184   1 09:54 pts/12   00:00:00 -bash
svetash+ 22849  673   0 27603 1016   1 11:35 pts/12   00:00:00 less
svetash+ 27790  673   1 38869 1876   0 11:51 pts/12   00:00:00 ps -F
```

5. Отримайте список процесів, запущених від імені вашого користувача із зазначенням наступного набору колонок: PID, TTY, PPID, STAT, NI, CMD

```
[svetashov_danilo@vpsj3IeQ ~]$ pstree svetashov_danilo | ps -o pid, tty, ppid, stat, ni, cmd
PID TTY      PPID STAT  NI CMD
 673 pts/12      657 Ss    0 -bash
22849 pts/12      673 T     0 less
29363 pts/12      673 R+    0 ps -o pid, tty, ppid, stat, ni, cmd
[svetashov_danilo@vpsj3IeQ ~]$
```

6. Отримайте список всіх сплячих процесів зі спрощеним набором колонок виведення таблиці процесів.

```
[svetashov_danilo@vpsj3IeQ ~]$ ps -Ao stat,pid,ppid,cmd | grep 'S '
S      2      0 [kthreadd]
S      6      2 [ksoftirqd/0]
S      7      2 [migration/0]
S      8      2 [rcu_bh]
S      9      2 [rcu_sched]
S     11      2 [watchdog/0]
S     12      2 [watchdog/1]
S     13      2 [migration/1]
S     14      2 [ksoftirqd/1]
S     18      2 [kdevtmpfs]
S     20      2 [khungtaskd]
S     35      2 [kswapd0]
S    114      2 [kauditd]
S    256      2 [scsi_eh_0]
S    258      2 [scsi_eh_1]
S    273      2 [jbd2/vda1-8]
S    520      1 /usr/sbin/chronyd
```

7. Отримайте список процесів, відсортованих по PID, і визначте:

```
top - 12:11:11 up 56 days, 19:14, 6 users, load average: 36.78, 35.84, 33.70
Tasks: total, running, sleeping, stopped, zombie
%Cpu(s): us, sy, ni, id, wa, hi, si, st
KiB Mem : total, free, used, buff/cache
KiB Swap: total, free, used, avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
31965	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kworker/0:2
31810	tsitkov+	20	0	115548	2132	1672	S	0.0	0.1	0:00.11	bash
31766	tsitkov+	20	0	163756	2668	1172	S	0.0	0.1	0:00.24	sshd
31723	kolesni+	20	0	113416	1504	1184	S	0.6	0.1	64:01.02	sh
31432	root	20	0	165892	6632	5120	S	0.0	0.4	0:00.34	sshd
31225	markovs+	20	0	130680	1644	1268	S	0.0	0.1	0:05.83	ping
30973	markovs+	20	0	130680	1640	1264	S	0.0	0.1	0:05.83	ping
30737	markovs+	20	0	130680	1644	1268	S	0.0	0.1	0:05.90	ping
30450	kelembe+	20	0	130680	1644	1264	S	0.0	0.1	0:03.82	ping
29760	apache	20	0	113480	3120	1592	S	0.0	0.2	0:00.14	httpd
29711	kalina_+	20	0	113416	1492	1176	S	0.0	0.1	0:08.99	kalina.sh
29606	root	20	0	163756	6584	5088	S	0.0	0.3	0:00.39	sshd
28853	shulyak+	20	0	115672	2196	1696	S	0.0	0.1	0:00.04	bash
28802	shulyak+	20	0	163756	2544	1052	S	0.0	0.1	0:00.07	sshd
28772	apache	20	0	113480	3096	1576	S	0.0	0.2	0:00.08	httpd
28743	kovach_+	20	0	130680	1640	1264	S	0.0	0.1	0:04.76	ping
28415	evchev_+	20	0	130680	1644	1264	S	0.0	0.1	0:20.00	ping
27717	belobro+	20	0	113416	1496	1176	S	0.0	0.1	78:33.99	belobrov.sh
27573	soft	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm
27572	soft	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm
27571	soft	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm
27570	emps	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm
27569	emps	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm
27568	emps	30	10	279912	5080	828	S	0.0	0.3	0:00.00	php-fpm

а. загальна кількість запущених процесів

а.28

b. кількість процесів, які виконуються

b.4

c. кількість сплячих процесів

c.24

8. Отримайте список процесів, відсортованих за використання процесора.

```
top - 12:10:33 up 56 days, 19:13, 6 users, load average: 37.00, 35.73, 33.59
Tasks: total, running, sleeping, stopped, zombie
%Cpu(s): us, sy, ni, id, wa, hi, si, st
KiB Mem : total, free, used, buff/cache
KiB Swap: total, free, used, avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
30316	tsitkov+	20	0	113288	1188	1012	D	13.6	0.1	0:29.68	sh
7647	bodnar_+	21	1	113284	1188	1012	R	11.9	0.1	155:24.34	sh
4014	bodnar_+	21	1	113284	1188	1012	R	11.6	0.1	157:12.83	sh
18360	nezhivi+	20	0	113284	1184	1008	R	9.3	0.1	169:03.79	sh
7548	oracle	20	0	113288	1188	1004	R	8.9	0.1	4892:03	test1.sh
7574	oracle	20	0	113288	1188	1004	R	8.9	0.1	4882:41	test2.sh
16180	bodnar_+	20	0	113284	1184	1008	R	8.9	0.1	160:21.24	sh
22242	barkar_+	20	0	113284	1188	1012	R	8.9	0.1	79:39.18	sh
10890	shulyak+	20	0	113284	1188	1012	R	8.6	0.1	0:35.25	sh
12346	grebeni+	20	0	113284	1188	1012	R	8.6	0.1	82:42.26	sh
18504	barkar_+	20	0	113284	1188	1012	R	8.6	0.1	88:59.75	sh
31002	sherbak+	20	0	115680	1120	592	R	8.6	0.1	44:10.95	bash
32231	sherbak+	20	0	115680	1120	584	R	8.6	0.1	43:33.87	bash
12903	nezhivi+	20	0	113284	1184	1008	R	8.3	0.1	152:07.82	sh
16975	shulyak+	25	5	113284	1184	1012	R	5.0	0.1	0:41.67	sh
11096	nezhivi+	25	5	113284	1184	1012	R	4.6	0.1	61:36.01	sh
13687	tsigank+	25	5	113284	1188	1012	R	4.6	0.1	9:22.99	sh
22598	barkar_+	25	5	113284	1188	1012	R	4.6	0.1	35:28.63	sh
29684	tsitkov+	25	5	113288	1188	1012	D	3.6	0.1	0:27.86	sh
16972	kalina_+	20	0	113416	1492	1176	S	0.7	0.1	0:07.20	kalina3.sh
24552	kolesni+	20	0	113416	1504	1184	S	0.7	0.1	65:11.69	sh
500	root	20	0	476444	1696	1000	S	0.3	0.1	782:50.02	NetworkManager
11578	nestere+	25	5	113416	1496	1176	S	0.3	0.1	69:59.44	nesterenko2.sh

## Завдання

### 3 Керування станами процесів

1. У поточному терміналі виконайте команду `ping localhost`, але не завершуйте її роботу.

```
[svetashov_danilo@vpsj3IeQ ~]$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.022 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.016 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.030 ms
```

2. Запустіть другий термінал доступу до Linux сервера.

3. У другому терміналі для команди `ping` отримаєте таблицю її процесу (колонки PID, STAT, CMD).

Можно так

```
[svetashov_danilo@vpsj3IeQ ~]$ pstree -p svetashov_danilo
sshd(22230)---bash(22250)---pstree(27742)

sshd(657)---bash(673)---less(22849)
                        |
                        |---ping(14620)
[svetashov_danilo@vpsj3IeQ ~]$ █

[svetashov_danilo@vpsj3IeQ ~]$ ps -p 14620 -o pid,stat,cmd
  PID STAT  CMD
14620 S+    ping localhost
[svetashov_danilo@vpsj3IeQ ~]$ █
```

---

Или так

```
[svetashov_danilo@vpsj3IeQ ~]$ ps -u svetashov_danilo -o pid,stat,cmd
  PID STAT  CMD
  657 S      sshd: svetashov_danilo@pts/12
  673 Ss     -bash
14620 S+    ping localhost
22230 S      sshd: svetashov_danilo@pts/16
22250 Ss     -bash
22849 T      less
32447 R+    ps -u svetashov_danilo -o pid,stat,cmd
```

4. У другому терміналі призупиніть виконання процес у команди ping

```

64 bytes from localhost: icmp_seq=19 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=20 ttl=64 time=0.017 ms
64 bytes from localhost: icmp_seq=21 ttl=64 time=0.026 ms
64 bytes from localhost: icmp_seq=22 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=23 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=24 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=25 ttl=64 time=0.027 ms
64 bytes from localhost: icmp_seq=26 ttl=64 time=0.018 ms
64 bytes from localhost: icmp_seq=27 ttl=64 time=0.018 ms
[1]+  Stopped                  ping localhost
[svetashov_danilo@vpsj3IeQ ~]$
64 bytes from localhost: icmp_seq=35 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=36 ttl=64 time=0.017 ms
64 bytes from localhost: icmp_seq=37 ttl=64 time=0.026 ms
64 bytes from localhost: icmp_seq=38 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=39 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=40 ttl=64 time=0.035 ms
64 bytes from localhost: icmp_seq=41 ttl=64 time=0.027 ms
64 bytes from localhost: icmp_seq=42 ttl=64 time=0.018 ms

--- localhost ping statistics ---
42 packets transmitted, 42 received, 0% packet loss, time 41257ms
rtt min/avg/max/mdev = 0.015/0.027/0.041/0.008 ms
[svetashov_danilo@vpsj3IeQ ~]$ ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
svetash+   673   0.0   0.1 115680  2184 pts/12   Ss   09:54   0:00 -bash
svetash+  3234   0.0   0.0 130680  1636 pts/16   S+   13:05   0:00 ping lc
svetash+  4014   0.0   0.0 155476  1876 pts/12   R+   13:05   0:00 ps -u
svetash+ 22250   0.0   0.1 115680  2188 pts/16   Ss   12:18   0:00 -bash
svetash+ 22849   0.0   0.0 110412  1016 pts/12   T    11:35   0:00 less
[svetashov_danilo@vpsj3IeQ ~]$ kill -19 3234
[svetashov_danilo@vpsj3IeQ ~]$

```

5. У першому терміналі отримайте список фонових процесів

```

[1]+  Stopped                  ping localhost
[svetashov_danilo@vpsj3IeQ ~]$

```

6. У другому терміналі відновіть виконання припиненого процесу

```

[svetashov_danilo@vpsj3IeQ ~]$ kill -18 3234
[3]-  Done                    jobs
[svetashov_danilo@vpsj3IeQ ~]$

```

7. У другому терміналі зупиніть виконання процесу команди ping

```

kill 3234

```

8. У першому терміналі запустіть команду ping в фоновому режимі так, щоб він не був автоматично зупинений навіть після закриття терміналу, з якого був запущений.

```

[svetashov_danilo@vpsj3IeQ ~]$ nohup ping localhost &
[2] 2438

```

9. Закрийте перший термінал

10. У другому терміналі для команди ping отримаєте таблицю її процесу (колонки PID, STAT, CMD). Зробіть висновок про стан процесу.



```
[svetashov_danilo@vpsj3IeQ ~]$ ps -p 2438 -o pid,stat,cmd
  PID STAT  CMD
  2438 S      ping localhost
[svetashov_danilo@vpsj3IeQ ~]$
```

Процес не зупинився, він перейшов в Sleeping

11. Завершіть роботу процесу.

```
[svetashov_danilo@vpsj3IeQ ~]$ kill 2438
```

## Завдання

### 4 Управління пріоритетами процесів

1. Створіть bash програму, що виконує операцію циклічного складання за формулою:  $x = x + n$ , де початкове значення  $x$  = кількість букв вашого прізвища,  $n$  кількість букв у вашому імені. Ім'я програми збігається з транслітерацією вашого прізвища з розширенням .sh, наприклад, ivanov.sh

```
#!/bin/bash

x=9
n=6

while [0]
do

x=$((x+n))
done
```

2. Запустіть bash програму у фоновому режимі.

Для запуску процесу у фоновому режимі в кінці командного рядка запуску програми додають символ &

```
[svetashov_danilo@vpsj3IeQ ~]$ sh svetashov.sh &
[2] 4885
```

3. Перегляньте таблицю процесів для запущеного процесу, пов'язаного з bash програмою, з урахуванням набором колонок: PID, PPID, STAT, NI,% CPU, CMD.

```
[svetashov_danilo@vpsj3IeQ ~]$ ps -o pid,ppid,stat,ni,%cpu,cmd -p 4885
  PID  PPID STAT  NI %CPU CMD
  4885   673  T      0  11.6 sh svetashov.sh
```

в stat має бути R

4. Виконайте команду призупинення запущеного процесу.

```
]$ kill -19 4885
^C
```



5. Ще раз перегляньте таблицю процесів для призупиненого процесу з урахуванням набором колонок: PID, PPID, STAT, NI, CPU, CMD. Зробіть висновки про його стан.

```
PID  PPID  STAT  NI  %CPU  CMD
4885  673  T      0   8.8  sh  svetashov.sh
[svetashov_danilo@vpsj3IeQ ~]$ █
```

– T (terminated)зупинений процес;

процес зупинився, але ps одразу не показує %CPU 0

6. Виконайте команду продовження виконання припиненого процесу.

7. Ще раз перегляньте таблицю процесів для процесу, який продовжив виконуватися, з урахуванням набором колонок: PID, PPID, STAT, NI, CPU, CMD. Зробіть висновки про його стан.

```
[svetashov_danilo@vpsj3IeQ ~]$ kill -18 4885
[svetashov_danilo@vpsj3IeQ ~]$ ps -o pid,ppid,stat,ni,%cpu,cmd -p 4885
PID  PPID  STAT  NI  %CPU  CMD
4885  673  R      0   6.9  sh  svetashov.sh
[svetashov_danilo@vpsj3IeQ ~]$ █
```

процес відновив роботу

8. Створіть два файли як символічні посилання на створену bash програму з іменами як ім'я поточного файлу з додаванням цифр 2 і 3, відповідно, наприклад: ivanov2.sh, ivanov3.sh

```
[svetashov_danilo@vpsj3IeQ ~]$ ln -s svetashov.sh svetashov2.sh
[svetashov_danilo@vpsj3IeQ ~]$ ln -s svetashov.sh svetashov3.sh
[svetashov_danilo@vpsj3IeQ ~]$ ls
accounts.csv          OC_lab_work_3.doc      svetashov3.sh
change.txt            OC_lab_work_3.pdf      svetashov.sh
lab_6                 OC_svetashov_danilo_lab_2.docx  svetashov_shostack
my_change_file.sh     Operating-System.-Laboratory-Work-1  table.csv
MyOSParam.sh          scriptcsv.sh            tree
nohup.out             svetashov2.sh           Наявне населення на 1 лютого 2021 р
[svetashov_danilo@vpsj3IeQ ~]$ █
```

9. Запустить два файли у фоновому режимі.

```
[svetashov_danilo@vpsj3IeQ ~]$ sh svetashov2.sh &
[3] 1746
[svetashov_danilo@vpsj3IeQ ~]$ sh svetashov3.sh &
[4] 4654
[svetashov_danilo@vpsj3IeQ ~]$ █
```

10. Перегляньте таблицю процесів для трьох запущених процесів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки за поточними значеннями NI та %CPU.

```
[svetashov_danilo@vpsj3IeQ ~]$ ps -o pid,ppid,stat,ni,%cpu,cmd -p 4885 1746 4654
PID  PPID  STAT  NI  %CPU  CMD
1746  673  R      0   8.9  sh  svetashov2.sh
4654  673  R      0   9.0  sh  svetashov3.sh
4885  673  R      0   8.4  sh  svetashov.sh
[svetashov_danilo@vpsj3IeQ ~]$ █
```

## 11. Зменшити пріоритет виконання одного з трьох процесів.

```
PID  PPID  STAT  NI  %CPU  CMD                                pid,ppid,stat,ni,COMMAND      pu,cmd,pr
1746   673  RN     5    6.6  sh  svetashov2.sh                    pid,ppid,stat,ni,sh          pu,cmd,pr
4654   673  R      0   10.3 sh  svetashov3.sh                    pid,ppid,stat,ni,sh          pu,cmd,pr
4885   673  R      0    9.1 sh  svetashov.sh                     pid,ppid,stat,ni,sh          pu,cmd,pr
[svetashov_danilo@vpsj3IeQ ~]$ renice -n -10 -p 1746
renice: failed to set priority for 1746 (process ID): Permission denied
[svetashov_danilo@vpsj3IeQ ~]$ █
```

---

## 12. Перегляньте таблицю процесів для трьох запущених файлів з урахуванням процесів для трьох запущених файлів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD.

```
PID  PPID  STAT  NI  %CPU  CMD                                pid,ppid,stat,ni,COMMAND      pu,cmd,pr
1746   673  RN     5    6.2 sh  svetashov2.sh                    pid,ppid,stat,ni,sh          pu,cmd,pr
4654   673  R      0   10.5 sh  svetashov3.sh                    pid,ppid,stat,ni,sh          pu,cmd,pr
4885   673  R      0    9.2 sh  svetashov.sh                     pid,ppid,stat,ni,sh          pu,cmd,pr
[svetashov_danilo@vpsj3IeQ ~]$ █
```

## Висновок:

Найважчим у даній роботі виявилась робота із пріоритетами.