

## Lab 3 470 OS

- 3
- Browser consumes memory (RAM)

```
john@john-VirtualBox:~$ grep MemTotal /proc/meminfo
MemTotal:      22066516 kB
john@john-VirtualBox:~$
```

- 3.

- 4.

```
john@john-VirtualBox:~$ ps aux
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.3  0.0 166584 11600 ?        Ss   16:34   0:00 /sbin/init sp
root           2  0.0  0.0      0      0 ?        S    16:34   0:00 [kthreadd]
root           3  0.0  0.0      0      0 ?        I<   16:34   0:00 [rcu_gp]
root           4  0.0  0.0      0      0 ?        I<   16:34   0:00 [rcu_par_gp]
root           5  0.0  0.0      0      0 ?        I<   16:34   0:00 [slub_flushwq]
root           6  0.0  0.0      0      0 ?        I<   16:34   0:00 [netns]
root           7  0.0  0.0      0      0 ?        I    16:34   0:00 [kworker/0:0-
root           8  0.0  0.0      0      0 ?        I<   16:34   0:00 [kworker/0:0H
root           9  0.2  0.0      0      0 ?        I    16:34   0:00 [kworker/u18:
root          10  0.0  0.0      0      0 ?        I<   16:34   0:00 [mm_percpu_wq]
root          11  0.0  0.0      0      0 ?        S    16:34   0:00 [rcu_tasks_ru
root          12  0.0  0.0      0      0 ?        S    16:34   0:00 [rcu_tasks_tr
root          13  0.0  0.0      0      0 ?        S    16:34   0:00 [ksoftirqd/0]
root          14  0.0  0.0      0      0 ?        I    16:34   0:00 [rcu_sched]
root          15  0.0  0.0      0      0 ?        S    16:34   0:00 [migration/0]
```

PID 1

- 6.

```
john@john-VirtualBox:~$ ps aux --sort=%mem |head
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           2  0.0  0.0      0      0 ?        S    16:34   0:00 [kthreadd]
root           3  0.0  0.0      0      0 ?        I<   16:34   0:00 [rcu_gp]
root           4  0.0  0.0      0      0 ?        I<   16:34   0:00 [rcu_par_gp]
root           5  0.0  0.0      0      0 ?        I<   16:34   0:00 [slub_flushwq]
root           6  0.0  0.0      0      0 ?        I<   16:34   0:00 [netns]
root           7  0.0  0.0      0      0 ?        I    16:34   0:00 [kworker/0:0-
events]
root           8  0.0  0.0      0      0 ?        I<   16:34   0:00 [kworker/0:0H
-events_highpri]
root           9  0.1  0.0      0      0 ?        I    16:34   0:00 [kworker/u18:
0-events_unbound]
root          10  0.0  0.0      0      0 ?        I<   16:34   0:00 [mm_percpu_wq]
john@john-VirtualBox:~$
```

PID 2 most memory

7.

```
john@john-VirtualBox:~$ gcc call.c -o call
john@john-VirtualBox:~$ ./call
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the parent process
am the child process
am the parent process
am the child process
am the parent process
am the child process
am the parent process
am the child process
am the parent process
am the child process
```

```
Firefox Web Browser
1 // ... .h>
2 #include <stdio.h>
3
4
5 int main()
6 {
7     int pid;
8     pid = fork();
9     for (int i = 0; i < 200; i++)
10     {
11         if (pid < 0)
12         {
13             printf("Fork failed \n");
14             return 1;
15         }
16
17         else if (pid == 0)
18         {
19             printf("I am the child process\n");
20
21         }
22         else
23         {
24             printf("I am the parent process \n");
25         }
26     }
27     return 0;
28 }
29
30
```

8.

```
john@john-VirtualBox: ~
john@john-VirtualBox:~$ gcc call.c -o call
john@john-VirtualBox:~$ ./call
I am the parent!
I am the Child!
john@john-VirtualBox:~$
```

```

1 #include <unistd.h>
2 #include <stdio.h>
3
4
5 int main()
6 {
7     if (fork() == 0)
8         printf("I am the Child! \n");
9     else
10        printf("I am the parent! \n");
11 }

```

9.

```

1 #include <unistd.h>
2 #include <stdio.h>
3
4
5 int main()
6 {
7     pid_t pid;
8     pid = fork();
9     if (pid == 0)
10    {
11        printf("pid in child=%d and parent =%d\n", getpid(),getppid());
12    }
13    else
14    {
15        printf("pid in parent=%d and child=%d\n", getpid(),pid);
16    }
17 }

```

```

john@john-VirtualBox:~$ gcc call.c -o call
john@john-VirtualBox:~$ ./call
pid in parent=5521 and child=5522
pid in child=5522 and parent =5521
john@john-VirtualBox:~$

```