

EX 1:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./print_pid &
[1] 1130987
tarjeisv@itstud:~/OS2022/labs/lab5$
The ID of this process is: 1130987
The ID of the parent process is: 1126574
^C
tarjeisv@itstud:~/OS2022/labs/lab5$ ./print_pid &
[2] 1130997
tarjeisv@itstud:~/OS2022/labs/lab5$
The ID of this process is: 1130997
The ID of the parent process is: 1126574
^C
tarjeisv@itstud:~/OS2022/labs/lab5$ ps -l
```

state	process id	parent process id
R	1130997	1126574
S	1130987	1126574
S	1130997	1126574
S	3069	1126574

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ps -l
```

TIME	CMD
00:00:00	bash
00:00:00	print_pid
00:00:00	print_pid
00:00:00	ps

The process with id 1126574 spawned the two processes and is the parent.

Kill the 2 processes:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./print_pid &
[1] 1131726
tarjeisv@itstud:~/OS2022/labs/lab5$
The ID of this process is: 1131726
The ID of the parent process is: 1126574
^C
tarjeisv@itstud:~/OS2022/labs/lab5$ ./print_pid &
[2] 1131728
tarjeisv@itstud:~/OS2022/labs/lab5$
The ID of this process is: 1131728
The ID of the parent process is: 1126574
^C
tarjeisv@itstud:~/OS2022/labs/lab5$ ps -l
```

TIME	CMD
00:00:00	bash
00:00:00	print_pid
00:00:00	print_pid
00:00:00	ps

```
tarjeisv@itstud:~/OS2022/labs/lab5$ kill 1131726
tarjeisv@itstud:~/OS2022/labs/lab5$ kill 1131728
[1]- Terminated ./print_pid
tarjeisv@itstud:~/OS2022/labs/lab5$
[2]+ Terminated ./print_pid
```

Before killing the processes:

0	S	3069	1131849	1126574	0	80	0	-	592	-	pts/16	00:00:00	print_pid
0	S	3069	1131855	1126574	0	80	0	-	592	-	pts/16	00:00:00	print_pid

After killing:

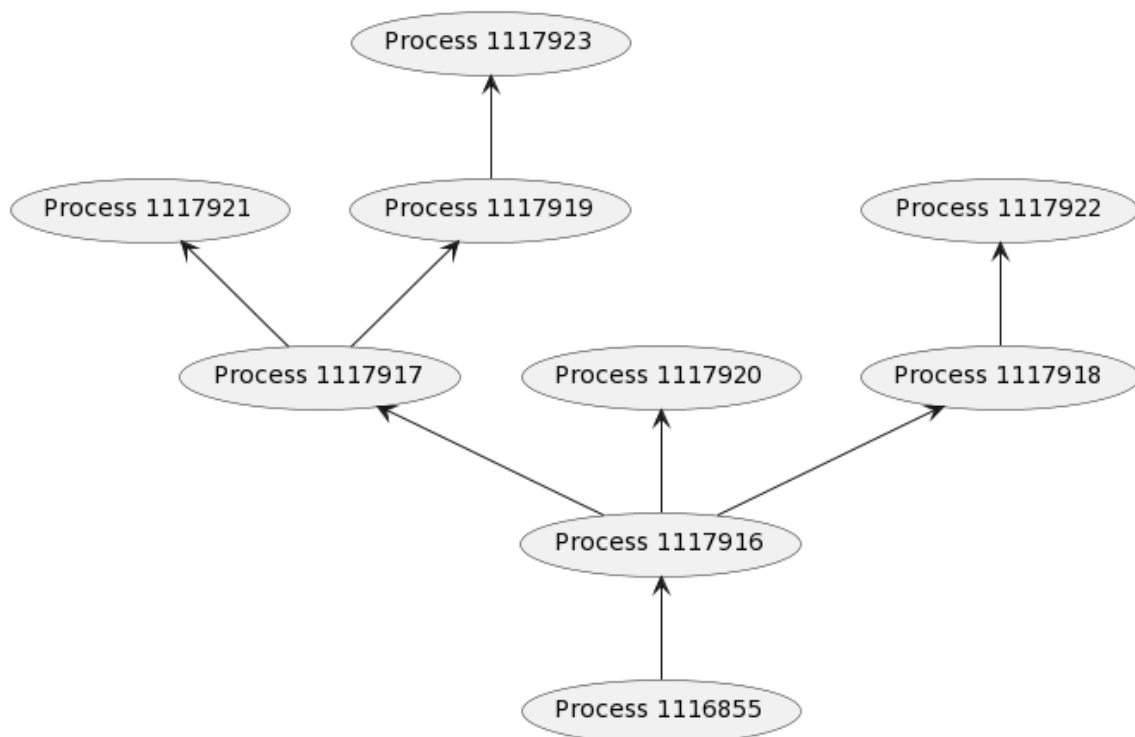
```
0 S 1524 1131775 1118071 0 80 0 - 3214 - pts/14 00:00:00 nano
0 S 2390 1131865 1131442 0 80 0 - 1620 - pts/31 00:00:00 nano
```

You can see that they are gone from the process table because they would be between these 2 processes.

Ex 2:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ gcc -o fork_ex3 fork_ex3.c
tarjeisv@itstud:~/OS2022/labs/lab5$ ./fork_ex3
Process 1117916's parent process ID is 1116855
Process 1117917's parent process ID is 1117916
Process 1117920's parent process ID is 1117916
Process 1117921's parent process ID is 1117917
Process 1117918's parent process ID is 1117916
Process 1117922's parent process ID is 1117918
Process 1117919's parent process ID is 1117917
Process 1117923's parent process ID is 1117919
```

online dia



If there are n fork(), 2^n processes will be created. It follows this pattern because every time you fork you split every process in 2.

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./fork_ex3
Process 1118513's parent process ID is 1116855
Process 1118514's parent process ID is 1118513
tarjeisv@itstud:~/OS2022/labs/lab5$ Process 1118518's parent process ID is 1
Process 1118516's parent process ID is 1
Process 1118517's parent process ID is 1
Process 1118515's parent process ID is 1
Process 1118520's parent process ID is 1
Process 1118519's parent process ID is 1
```

The process dies and its “kids” get inherited by systemd

Ex3:

LS:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./exec1
exec1    fork_ex1.c  fork_ex3.c  no_wait.c  print_pid  wait.c
exec1.c  fork_ex3    lab5.pdf   output    print_pid.c  waitpid.c
```

Pdw:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./exec1
/home/tarjeisv/OS2022/labs/lab5
```

Ps:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./exec1
  PID TTY          TIME CMD
1116855 pts/4    00:00:00 bash
1120082 pts/4    00:00:00 ps
```

Cat:

```
tarjeisv@itstud:~/OS2022/labs/lab5$ ./exec1
```

Comment:

There was complications some with pushing the assignment, i ended up deleting the whole lab5 directory, this had some unforeseen consequences and i couldnt pull it back. I have tried to remake all the files manually, but i might have missed some.

