

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./exec_demo
[child] about to exec ls -l
total 124
-rw-rw-rw- 1 codespace codespace 12457 Sep  9 19:31 CMakeCache.txt
drwxrwxrwx+ 9 codespace codespace  4096 Sep  9 20:02 CMakeFiles
-rw-rw-rw- 1 codespace codespace 10678 Sep  9 20:02 Makefile
-rw-rw-rw- 1 codespace codespace  1626 Sep  9 19:31 cmake_install.cmake
-rwxrwxrwx 1 codespace codespace 16200 Sep  9 19:32 exec_demo
-rwxrwxrwx 1 codespace codespace 16360 Sep  9 19:32 fork_demo
-rwxrwxrwx 1 codespace codespace 16448 Sep  9 19:32 pipe_demo
-rwxrwxrwx 1 codespace codespace 16464 Sep  9 20:02 pipeline_demo
-rwxrwxrwx 1 codespace codespace 16304 Sep  9 20:02 zombie_demo
[parent] child finished exec

```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./fork_demo
[parent] created child PID=75580
[child] PID=75580, PPID=75579
[child] exiting with code 42
[parent] child 75580 exited with status 42

```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./pipe_demo
5

```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./pipeline_demo
[parent] forked child1 with PID=75753 and child2 with PID=75754
[child2] PID=75754 about to exec grep .c
CMakeCache.txt
cmake_install.cmake
exec_demo
[parent] finished waiting for both children

```

```

○ @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./zombie_demo
Parent PID=76364, child PID=76365
Check 'ps -el | grep Z' in this terminal or another. Parent sleeping forever...
Child exiting: PID=76365
_

```

```

● @kitzki →/workspaces/lab3.1 (main) $ ps -l

```

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
4	S	1000	75813	2110	0	80	0	-	3985	do_wai	pts/0	00:00:00	bash
0	R	1000	76409	75813	0	80	0	-	2872	-	pts/0	00:00:00	ps

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch1_single_fork
Hello from child (PID=110686, PPID=110685)
child 110686 exited with status 7
● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch2_multi_child 5
Parent creating 5 children
Child 0 starting
Child 2 starting
Child 3 starting
Child 1 starting
Child 4 starting
Parent waiting for all children to finish...
Child 0 (PID 110904) finished with exit code 1
Child 1 (PID 110905) finished with exit code 2
Child 2 (PID 110906) finished with exit code 3
Child 4 (PID 110908) finished with exit code 5
Child 3 (PID 110907) finished with exit code 4
All children have finished

```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch3_exec_ls
total 320
drwxrwxrwx+ 3 codespace codespace 4096 Sep  9 21:03 .
drwxrwxrwx+ 6 codespace codespace 4096 Sep  9 19:21 ..
-rw-rw-rw-  1 codespace codespace 12457 Sep  9 19:31 CMakeCache.txt
drwxrwxrwx+ 19 codespace codespace 4096 Sep  9 21:03 CMakeFiles
-rw-rw-rw-  1 codespace codespace 26546 Sep  9 21:02 Makefile
-rwxrwxrwx  1 codespace codespace 16544 Sep  9 20:34 ch10_pool
-rwxrwxrwx  1 codespace codespace 16320 Sep  9 20:23 ch1_single_fork
-rwxrwxrwx  1 codespace codespace 16384 Sep  9 20:34 ch2_multi_child
-rwxrwxrwx  1 codespace codespace 16176 Sep  9 20:34 ch3_exec_ls
-rwxrwxrwx  1 codespace codespace 16272 Sep  9 21:03 ch4_exec_worker
-rwxrwxrwx  1 codespace codespace 16280 Sep  9 20:34 ch5_exec_examples
-rwxrwxrwx  1 codespace codespace 16392 Sep  9 20:34 ch6_pipe_sum
-rwxrwxrwx  1 codespace codespace 16464 Sep  9 20:34 ch7_pipeline
-rwxrwxrwx  1 codespace codespace 16344 Sep  9 20:34 ch8_wait_nonblock
-rwxrwxrwx  1 codespace codespace 16352 Sep  9 20:34 ch9_zombie
-rw-rw-rw-  1 codespace codespace  1626 Sep  9 19:31 cmake_install.cmake
-rwxrwxrwx  1 codespace codespace 16200 Sep  9 19:32 exec_demo
-rwxrwxrwx  1 codespace codespace 16360 Sep  9 19:32 fork_demo
-rwxrwxrwx  1 codespace codespace 16448 Sep  9 19:32 pipe_demo
-rwxrwxrwx  1 codespace codespace 16464 Sep  9 20:02 pipeline_demo
-rwxrwxrwx  1 codespace codespace 16304 Sep  9 20:02 zombie_demo
Parent: child finished executing ls -la

```

```
● @kitzki →/workspaces/lab3.1/os-lab/challenges (main) $ gcc -o worker worker.c
● @kitzki →/workspaces/lab3.1/os-lab/challenges (main) $ gcc -o ch4_exec_worker ch4_exec_worker.c
● @kitzki →/workspaces/lab3.1/os-lab/challenges (main) $ ./ch4_exec_worker
Main program: Preparing to exec worker program
Parent: Waiting for worker to complete...
Child: Setting up environment and arguments for worker
Child: Executing worker with custom environment (MYVAR=hello)
=== Worker Program Output ===
Worker received 4 arguments:
  argv[0] = "./worker"
  argv[1] = "arg1"
  argv[2] = "arg2"
  argv[3] = "hello world"
Environment variable MYVAR = "hello"
Worker program completed successfully.
Parent: Worker completed with exit status 0
Parent: Demonstration completed
```

```
● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch5_exec_examples
Demonstrating execl vs execv with two child processes

Parent waiting for Child A...
Child B: Using execv with argument array
Child A: Using execl with individual arguments
one two
one two
Child A completed

Parent waiting for Child B...
Child B completed

Both children have finished. Both should have printed 'one two'
```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch6_pipe_sum
Parent creating pipe for communication with child
Parent: Writing numbers 1-10 to pipe...
Parent: Writing number 1
Child: Reading numbers from pipe...
Child: Read number 1
Parent: Writing number 2
Parent: Writing number 3
Parent: Writing number 4
Parent: Writing number 5
Parent: Writing number 6
Parent: Writing number 7
Parent: Writing number 8
Parent: Writing number 9
Parent: Writing number 10
Child: Read number 2
Child: Read number 3
Child: Read number 4
Child: Read number 5
Child: Read number 6
Child: Read number 7
Child: Read number 8
Child: Read number 9
Child: Read number 10
Child: Read 10 numbers, sum = 55
Parent: Child has finished processing all numbers

```

```

● @kitzki →/workspaces/lab3.1 (main) $ ps -l

```

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
4	S	1000	75813	2110	0	80	0	-	3985	do_wai	pts/0	00:00:00	bash
0	R	1000	112982	75813	0	80	0	-	2872	-	pts/0	00:00:00	ps

```

○ @kitzki →/workspaces/lab3.1 (main) $ 

```

```

● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch7_pipeline "\.c$"
Creating pipeline: ls | grep \.c$

Pipeline completed.
ls exit status: 0
grep exit status: 1

```

```
● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch8_wait_nonblock
Parent spawning 3 children with sleep times: 1s, 2s, 3s
Parent: Created child 1 with PID 112456
Parent: Created child 2 with PID 112457
Parent: Created child 3 with PID 112458

Parent: All children created, now polling with WNOHANG...
Parent: No children finished yet, continuing other work...
Child 1 (PID 112456): Starting, will sleep for 1 second(s)
Child 2 (PID 112457): Starting, will sleep for 2 second(s)
Child 3 (PID 112458): Starting, will sleep for 3 second(s)
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Child 1 (PID 112456): Finished sleeping, exiting
Parent: Child PID 112456 finished (exit code 1) [1/3 completed]
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
```

```
Parent: Child PID 112456 finished (exit code 1) [1/3 completed]
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Child 2 (PID 112457): Finished sleeping, exiting
Parent: Child PID 112457 finished (exit code 2) [2/3 completed]
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Parent: No children finished yet, continuing other work...
Child 3 (PID 112458): Finished sleeping, exiting
Parent: Child PID 112458 finished (exit code 3) [3/3 completed]

Parent: All children have finished. Parent is not blocked during waiting!
```



```
● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch9_zombie
```

```
Parent PID: 112919
```

```
Creating a child process that will become a zombie...
```

```
Parent: Created child with PID 112920
```

```
=== ZOMBIE DEMONSTRATION ===
```

```
The child has exited but parent hasn't called wait() yet.
```

```
In another terminal, run: ps -l | grep 112920
```

```
You should see the child process with state 'Z' (zombie).
```

```
Also try: ps -eo pid,ppid,state,comm | grep 112920
```

```
Parent sleeping for 10 seconds to keep zombie alive...
```

```
Parent: 10 seconds remaining (zombie should be visible in ps)
```

```
Child PID: 112920, Parent PID: 112919
```

```
Child: Exiting immediately to become a zombie
```

```
Parent: 9 seconds remaining (zombie should be visible in ps)
```

```
Parent: 8 seconds remaining (zombie should be visible in ps)
```

```
Parent: 7 seconds remaining (zombie should be visible in ps)
```

```
Parent: 6 seconds remaining (zombie should be visible in ps)
```

```
Parent: 5 seconds remaining (zombie should be visible in ps)
```

```
Parent: 4 seconds remaining (zombie should be visible in ps)
```

```
Parent: 3 seconds remaining (zombie should be visible in ps)
```

```
Parent: 2 seconds remaining (zombie should be visible in ps)
```

```
Parent: 1 seconds remaining (zombie should be visible in ps)
```

```
=== ZOMBIE CLEANUP ===
```

```
Parent: Now calling wait() to clean up the zombie...
```

```
● @kitzki →/workspaces/lab3.1/os-lab/build (main) $ ./ch10_pool 3 file1 file2 file3 file4 file5
```

```
Worker Pool Manager: Processing 5 tasks with max 3 workers
```

```
Manager: Started worker 1 (PID 113407) for task 'file1' [1/3 active]
```

```
Worker 1 (PID 113407): Starting task 'file1'
```

```
Manager: Started worker 2 (PID 113408) for task 'file2' [2/3 active]
```

```
Worker 2 (PID 113408): Starting task 'file2'
```

```
Manager: Started worker 3 (PID 113409) for task 'file3' [3/3 active]
```

```
Worker 3 (PID 113409): Starting task 'file3'
```

```
Worker 1 (PID 113407): Completed task 'file1' after 3 seconds
```

```
Worker 2 (PID 113408): Completed task 'file2' after 3 seconds
```

```
Manager: Worker PID 113407 finished [1/5 tasks completed, 2/3 active]
```

```
Worker 3 (PID 113409): Completed task 'file3' after 3 seconds
```

```
Manager: Started worker 4 (PID 113461) for task 'file4' [3/3 active]
```

```
Manager: Worker PID 113408 finished [2/5 tasks completed, 2/3 active]
```

```
Worker 4 (PID 113461): Starting task 'file4'
```

```
Manager: Started worker 5 (PID 113462) for task 'file5' [3/3 active]
```

```
Manager: Worker PID 113409 finished [3/5 tasks completed, 2/3 active]
```

```
Worker 5 (PID 113462): Starting task 'file5'
```

```
Worker 4 (PID 113461): Completed task 'file4' after 3 seconds
```

```
Worker 5 (PID 113462): Completed task 'file5' after 3 seconds
```

```
Manager: Worker PID 113461 finished [4/5 tasks completed, 1/3 active]
```

```
Manager: Worker PID 113462 finished [5/5 tasks completed, 0/3 active]
```

```
Manager: All 5 tasks completed successfully!
```

```
Manager: Worker pool demonstration finished.
```