# Programming #1 : fork()

- Child Process 를 생성한다 (fork())
- Parent Process와 Child Process 는 각각 value를 0부터 200까지 출력한다.
- 출력형태 : This line is form pid #, value = #

```
minsub@ubuntu:~/Desktop/forkt/report$ fork_report1
This line is from pid 3052, value = 1
This line is from pid 3052, value = 2
This line is from pid 3052, value = 3
This line is from pid 3052, value = 4
This line is from pid 3052, value = 5
This line is from pid 3052, value = 6
This line is from pid 3052, value = 7
This line is from pid 3052, value = 8
This line is from pid 3052, value = 9
This line is from pid 3052, value = 10
This line is from pid 3053, value = 1
This line is from pid 3052, value = 11
This line is from pid 3052, value = 12
This line is from pid 3053, value = 2
```

# Programming #2 : fork()

- Child Process 를 생성한다 (fork())
- Child Process가 생성 된 경우는 ChildProcess() 함수를 호출하여 값을 0부터 200까지 증가
- Parent Process가 생성 된 경우는 ParentProcess() 함수를 호출하여 값을 0부터 200까지 증가
- 출력형태
  - Parent 일 경우  This line is from parent, Parent_PID #, value = #
  - Child 일 경우 This line is from child, Child_PID #, value = #

```
This line is from parent, Parent_PID = 3448 , value = 17
This line is from parent, Parent_PID = 3448 , value = 18
This line is from parent, Parent_PID = 3448 , value = 19
   This line is from child, Child_PID = 3449 , value = 1
This line is from parent, Parent_PID = 3448 , value = 20
This line is from parent, Parent_PID = 3448 , value = 21
   This line is from child, Child_PID = 3449 , value = 2
This line is from parent, Parent_PID = 3448 , value = 22
   This line is from child, Child_PID = 3449 , value = 3
This line is from parent, Parent_PID = 3448 , value = 23
   This line is from child, Child_PID = 3449 , value = 4
This line is from parent, Parent_PID = 3448 , value = 24
   This line is from child, Child_PID = 3449 , value = 5
```

# Programming #3 : fork()

- Child Process 를 생성한다 (fork())
- Parent Process와 Child Proecess는 각자 자신이 수행 되고 있음을 표시한다.
- Parent Process와 Child Process는 각자 자신의 PID를 출력한다.
- Parent Process와 Child Process 는 10초동안 sleep한다.
- Child Process는 Sleep이후 exec 명령어를 통해 ls –al 명령을 수행한다.

- 아래의 두 가지 상태를 확인한 화면과 차이점에 대해 제출
  - Parent Process에서 자식을 기다리는 wait를 한 경우.
  - Parent Process에서 자식을 기다리는 wait를 하지 않은 경우.

# Programming #3 : fork() 출력형태

```
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
Parent Run


PARENT - MY PID :3823   child's PID : 3824

Child Run

CHILD-MY PID : 3824 parent's PID : 3823
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3823  0.0  0.0   4192   356 pts/14   S    16:39   0:00 fork_report3_nowait
minsub    3824  0.0  0.0   4192    88 pts/14   S    16:39   0:00 fork_report3_nowait
minsub    3825  0.0  0.0  22640  1308 pts/14   R+   16:39   0:00 ps -u
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$ total 40
drwxrwxr-x 2 minsub minsub 4096 Mar 18 16:39 .
drwxrwxr-x 3 minsub minsub 4096 Mar 18 16:07 ..
-rwxrwxr-x 1 minsub minsub 8874 Mar 18 16:39 fork_report3_nowait
-rw-r--r-- 1 minsub minsub  738 Mar 18 16:33 fork_report3_nowait.c
-rwxrwxr-x 1 minsub minsub 8871 Mar 18 14:15 fork_report3_wait
-rw-r--r-- 1 minsub minsub  894 Mar 18 14:08 fork_report3_wait.c
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3823  0.0  0.0   4192   356 pts/14   S    16:39   0:00 fork_report3_nowait
minsub    3824  0.0  0.0      0     0 pts/14   Z    16:39   0:00 [ls] <defunct>
minsub    3827  0.0  0.0  22640  1312 pts/14   R+   16:39   0:00 ps -u
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
 Parent END

End
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3828  0.0  0.0  22640  1312 pts/14   R+   16:39   0:00 ps -u
[1]+  Done                    fork_report3_nowait
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
```

No wait

```
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
Parent Run

PARENT_MY PID :3890   child's PID : 3891


Child Run
CHILD_MY PID : 3891 parent's PID : 3890
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3890  0.0  0.0   4192   356 pts/14   S    16:42   0:00 fork_report3_wait
minsub    3891  0.0  0.0   4192    88 pts/14   S    16:42   0:00 fork_report3_wait
minsub    3892  0.0  0.0  22640  1308 pts/14   R+   16:42   0:00 ps -u
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$ total 40
drwxrwxr-x 2 minsub minsub 4096 Mar 18 16:41 .
drwxrwxr-x 3 minsub minsub 4096 Mar 18 16:07 ..
-rwxrwxr-x 1 minsub minsub 8874 Mar 18 16:39 fork_report3_nowait
-rw-r--r-- 1 minsub minsub  738 Mar 18 16:33 fork_report3_nowait.c
-rwxrwxr-x 1 minsub minsub 8922 Mar 18 16:41 fork_report3_wait
-rw-r--r-- 1 minsub minsub  862 Mar 18 16:41 fork_report3_wait.c
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3890  0.0  0.0   4192   356 pts/14   S    16:42   0:00 fork_report3_wait
minsub    3893  0.0  0.0  22640  1308 pts/14   R+   16:42   0:00 ps -u
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
Parent END

END
ps -u
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
minsub    2899  0.0  0.0  27256  4388 pts/14   Ss   15:21   0:00 bash
minsub    3303  0.0  0.0  27212  4336 pts/0    Ss+  15:47   0:00 bash
minsub    3895  0.0  0.0  22640  1308 pts/14   R+   16:42   0:00 ps -u
[1]+  Done                    fork_report3_wait
minsub@ubuntu:~/Desktop/forkt/report/fork_report3$
```

Wait

# Programming #4 : fork()

- Child Process 10개를 생성한다 (fork(), for loop)
- 각 프로세스는 자신의 프로세스 식별자와 몇 번째 출력인지, 그리고 현재 시간을 특정파일에 기록한다. (ms 단위까지)
- 출력파일명 : out_1
- Child Process들은 자신들의 값을 100번 출력하고 종료한다.
- Child Process는 자신의 값을 한번 출력 한 후 1ms 대기시간을 가진다.
- Parent Process는 Child Process의 일이 종료 될 때 까지 기다리고, 종료되면 Child Process들의 프로세스 식별자를 출력한다.
- 출력형태
  - Child has finished : PID[#]

# Programming #4 : fork() 출력형태

```
minsub@ubuntu:~/Desktop/forkt/report$ fork_report4
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
Parent PID : 18202
minsub@ubuntu:~/Desktop/forkt/report$ child has finished : PID[18204]
child has finished : PID[18203]
child has finished : PID[18207]
child has finished : PID[18209]
child has finished : PID[18205]
child has finished : PID[18212]
child has finished : PID[18206]
child has finished : PID[18211]
child has finished : PID[18208]
child has finished : PID[18210]
```

fork() 실행 후 화면

```
minsub@ubuntu:~/Desktop/forkt/report$ cat out_1.txt
[Process 1 #0, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 983970mil
[Process 1 #1, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 985011mil
[Process 1 #2, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 986830mil
[Process 1 #3, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 988828mil
[Process 1 #4, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 989985mil
[Process 1 #5, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 991846mil
[Process 1 #6, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 993832mil
[Process 1 #7, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 995286mil
[Process 1 #8, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 996832mil
[Process 1 #9, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 998831mil
[Process 1 #10, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 1s. 837mil
[Process 1 #11, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 2844mil
[Process 1 #12, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 4835mil
[Process 1 #13, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 6873mil
[Process 1 #14, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 8833mil
[Process 1 #15, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 10285mil
[Process 1 #16, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 11861mil
[Process 1 #17, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 13977mil
[Process 1 #18, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 15927mil
[Process 1 #19, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 17079mil
[Process 1 #20, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 18853mil
[Process 1 #21, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 20833mil
[Process 1 #22, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 22860mil
[Process 1 #23, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 24832mil
[Process 1 #24, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 26832mil
[Process 1 #25, PID(18204)] current time : 2016y. 3m. 29d. 17h. 1m. 2s. 28832mil
```

out_1.txt