

# Programming Multiple fork

## Code:

```
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>

int main()
{
    for (int i = 0; i < 3; i++)
    {
        int pid = fork();

        if (pid == 0)
        {
            printf("child process. PPID = %d, PID= %d\n", getppid(), getpid());
            exit(0);
        }
        else
        {
            printf("Parent process. PPID= %d\n", getppid());
        }
    }

    return 0;
}
```

## Output Screenshot:

```
f12r@fahim: /media/f12r/Projects/CSE323.2-NVA/Programming multiple fork$ gcc -o fork_system_call fork_system_call.c
(base) f12r@fahim: /media/f12r/Projects/CSE323.2-NVA/Programming multiple fork$ ./fork_system_call
Parent process. PPID= 13315
Parent process. PPID= 13315
child process. PPID = 13416, PID= 13417
Parent process. PPID= 13315
child process. PPID = 13416, PID= 13418
child process. PPID = 13416, PID= 13419
(base) f12r@fahim: /media/f12r/Projects/CSE323.2-NVA/Programming multiple fork$
```

## Fork Tree Diagram:

### fork tree

