New York University
Tandon School of Engineering
Department of Computer Science and Engineering

Introduction to Operating Systems
Fall 2025

Assignment 4
(10 points)

A) (2 points) If you create a main() routine that calls fork() three times, i.e. if it includes the following code:

```
pid_t x=-11, y=-22, z=-33;
x = fork();
if(y>0) = fork();
z = fork();
```

Assuming all `fork()` calls succeed, draw a process tree similar to that of Fig. 3.8 (page 116) in your text book, clearly indicating the values of x, y and z for each process in the tree (i.e. whether 0,-11,-22,-33, or larger than 0).

Note that the process tree should only have one node for each process and thus the number of nodes should be equal to the number of processes.

The process tree should be a snapshot just after all forks completed but before any process exists.
Each line/arrow in the process tree diagram shall represent a creation of a process, or alternatively a parent/child relationship.

B) (4 points) Write a subroutine that creates the process tree shown below and returns 0 for parent, 1 for 1st child, 2 for second child, and 3 for third child.:

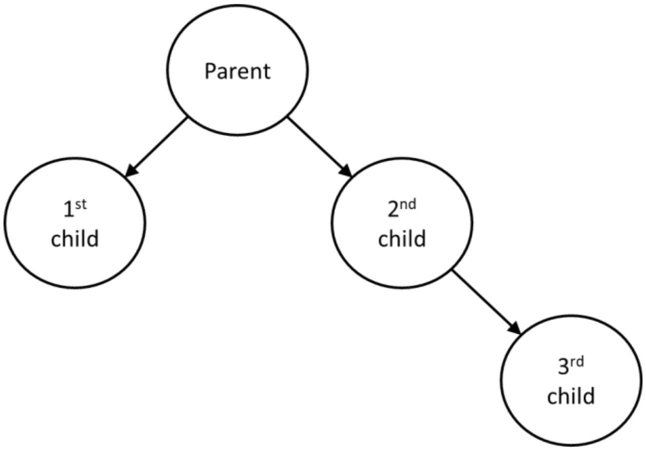

Fig. (1) - Process tree

C) (4 points) Write a program whose main routine obtains two parameters from the user, *n* and *d* (i.e., passed to your program when it was invoked from the shell, `n>0`) and creates a child process.
The child process shall then create **and print** an arithmetic sequence of length *n* and whose elements are of type `int`, such that each element has a value of *kd*, where *k* is the element number (`0` to *n*`-1`). For example, if *n*`=5` and *d*`=2`, the sequence shall be 0,2,4,6 and 8.

The parent waits for the child to exit and then prints 2 additional elements of the sequence, i.e. the total number of elements printed by the child and the parent is `n+2`.

Do not use IPC in your solution to this problem (i.e. neither shared memory nor message passing).

What to submit to gradescope:

Please submit the following files individually:

1) Source file(s) with appropriate comments.
   The naming should be similar to "**lab#_$.c**" (**#** is replaced with the assignment number and **$** with the question number within the assignment, e.g. lab4_b.c, for lab 4, question b OR lab5_1a for lab 5, question 1a).
2) A single pdf file (for images + report/answers to questions), named **"lab#.pdf"** (**#** is replaced by the assignment number), containing:
   - Screen shot(**s**) of your terminal window showing the current directory, the command used to compile your program, the command used to run your program and the output of your program.
3) Your Makefile, if any. This is applicable only to kernel modules.

RULES:

- You shall **use kernel version 4.x.x or above**. You shall not use kernel version 3.x.x.
- You may consult with other students about GENERAL concepts or methods but copying code (or code fragments) or algorithms is NOT ALLOWED and is considered cheating (whether copied form other students, the internet or any other source).
- If you are having trouble, please ask your teaching assistant for help.
- You must submit your assignment prior to the deadline.