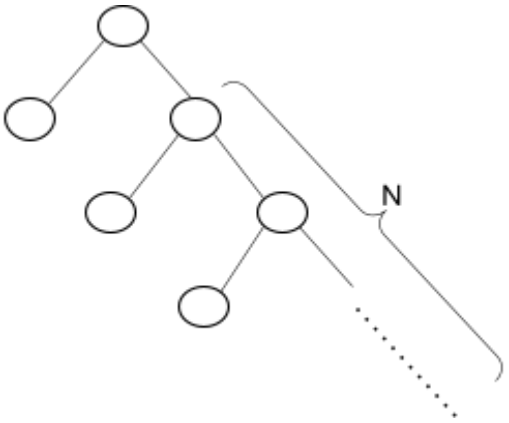
Computer Operating Systems
Assignment 1

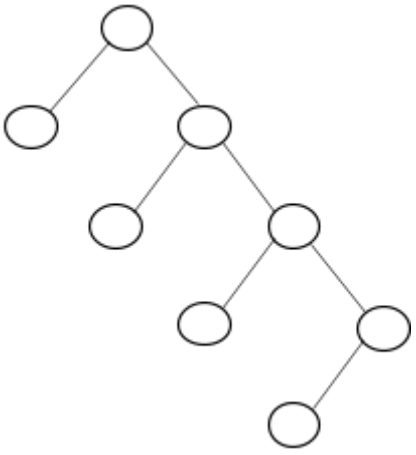
Deadline April 2, 2023 @23:59

In this assignment, you are tasked to use the "fork()" system call to create a specific process hierarchy given in the figure below. Getting better at handling fork system calls provides a better understanding of process management in operating systems which is quite useful for parallel programming and resource management.

Question 1



The left side of the tree is always the same and the right part of the given process tree is parametric and its depth changes by the N value, which can take any positive integer. For example if N = 3, the code must create the following process hierarchy.

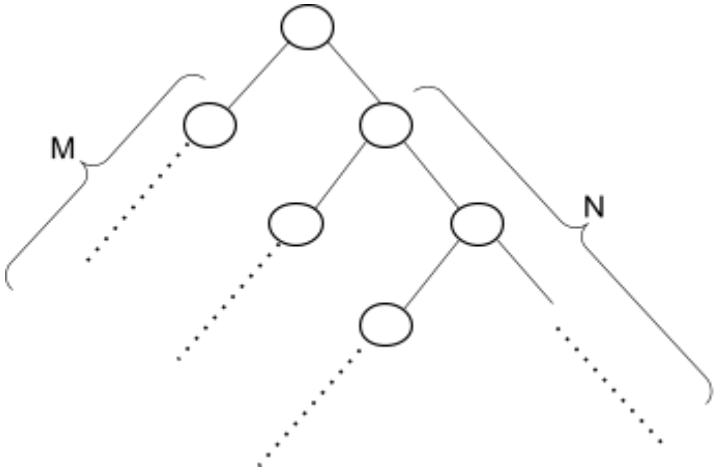


Considering these, do following tasks:

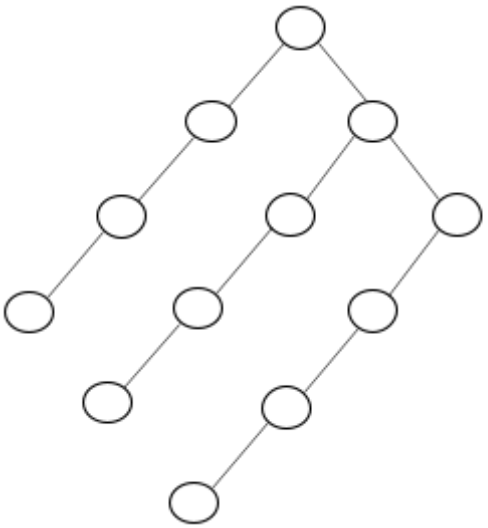
1. Write a code that implements the given parametric topology of the process tree. Your code should use the given N value to create this parametric process tree.
2. Derive an expression on how many of the created processes can be identified as parent processes depending on value N.

Question 2

Now we will extend the solution and make left subtrees' depths parametric too. As in the figure below, the depths of each left subtree changes with positive integer M.



Here is another example with N = 2 and M = 3



As a programming tip, you can use recursive programming to divide right and left subtree creation. Considering these, do following tasks:

3. Extend your existing solution to adjust the depths of left subtrees. Your code should use the given M value.
4. Derive an expression on how many of the created processes can be identified as parent processes depending on N and M value.

Submission Rules

- Submit (1) your report that explains your code and contains the answers for the given question to Ninova as a pdf file.
- Submit (2) the C files.
- Create a **makefile** of your code and submit (3) it.
- Prepare your reports in LaTeX format to get full marks. You can use the following template: https://www.overleaf.com/read/dqshxtrpthgh
- Every child process must terminate and they must print their PID and their parent's PID to console.
- Your report must not exceed 5 pages.
- Any form of **plagiarism** will not be tolerated. You must solve each question by yourself.
- You must implement your solution in the C programming language and it should work on a linux environment.