Two PCB implementations

Objective:

Study of performance of two different PCB implementations

Description:

Write a program that simulates creation and destruction of PCBs for operating system with two different implementations that we studied:

- With linked list
- Without linked list

1. With linked list:
   Each PCB has only two fields:
   a. Parent – index
   b. Children – pointer to a linked list which contains the PCB index
2. Without linked list:
   Each PCB has only four fields:
   a. Parent
   b. First child
   c. Younger sibling
   d. Older sibling

<u>Other details:</u>

1. Have PCBs as an array of size N
2. Array index is a process ID
3. Assume that only one process exists initially, that is, PCB[0].
4. Implement Create(p) function that simulates creating a new child process of a process p. Allocate next available index as child process ID. Initialize the values of the fields.
   a. New process's parent is set to p.
   b. New process's children list is empty. (linked list)
   c. Set to null value for other fields. (without linked list)
5. Implement Destroy(p) function that simulates destroying all processes and their descendants recursively. You are to keep up with all memory management. For

instance, deallocate the element from the linked list. Indicate that PCB array element free.

6. Write a test program that performs a long series of process creations and destructions for each implementation separately: 100~200 creations and 20~50 destructions (How many processes are normally running on a laptop?)
7. Start time at the beginning of test and end at the end of test for each implementation to measure how long it takes to perform them all

Output your simulation results in meaningful way and explain what you observe from them. For example, you may explain why one implementation performs better than the other one

What to submit:

- Your code
- Simulation test plan
- Outputs (in meaningful ways)
- Write-up (explain your observations)