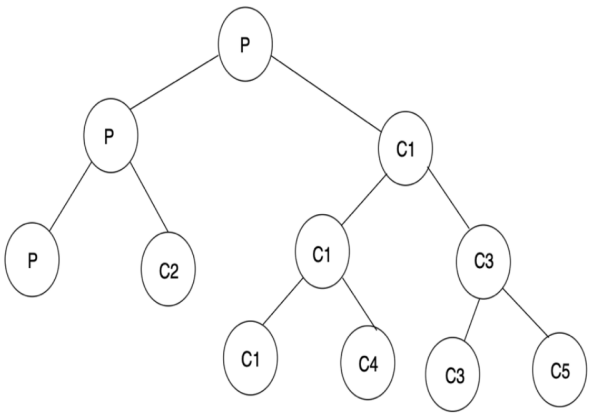
Lab Tasks

Do the following tasks in the allocated time.

Task 1 [5 minutes]

Consider the following tree. Get the process id of the process “P” while being in the “C5” process.



Task 2 [20 minutes]

Write a program that creates multiple child processes, each with its unique task. The parent process should wait for all child processes to finish before printing a final message.

Task 3 [20 minutes]

Write a program that creates multiple child processes. Each child process performs a different task and exits with a specific exit status. The parent process should use wait to collect the exit statuses of all child processes and display them.

Task 4 [25 minutes]

Create a program that intentionally leaves a child process in a zombie state. Add code to your program to demonstrate how to prevent the creation of zombie processes and clean them up.

Task 5 [25 minutes]

Create a program that simulates resource allocation and deallocation using child processes. Each child process represents a resource allocation request, and the parent process manages a pool of resources. Use wait to handle resource deallocation and allocation requests.

Hint:
You can create a global variable representing number of available resources. A request variable can be created that represents number of requested resources.