Quiz#2

Time: **9 +1 mins**

Solution

Q#1.

a. What command would you use to view the complete list of processes running on your system? Explain the purpose of at least three columns from the output. (2 mins)

- Command: ps au
- User: The user who owns the process.
- PID: The process ID, which is a unique identifier for each process.
- STAT: The current state of the process (e.g., R for running, S for sleeping, Z for zombie).

b. You ran the command **ps au** and noticed a process with the state marked as Z. What does this state indicate? (1 mins)

- A **Z** in the STAT column indicates a **Zombie process**, which is a process that has completed execution but still has an entry in the process table. This occurs because the parent has not yet read the exit status of the child

Q#2.

a. If you call **fork()** once in your code, how many processes will be created? How many will be created if you call fork() twice consecutively? Can you draw the tree for that? (3 mins)

- If you call fork() once, 2 processes are created: the parent and the child.
- If you call fork() twice, 4 processes are created. This happens because the first fork() creates 2 processes, and then both the parent and child processes each call fork() again, resulting in 4 processes in total.
- $\mathbf{2^n}$

b. In a system call **fork()**, what is the significance of the return value, and how can you differentiate between the parent and child process based on it? (3 mins)

- The fork() system call returns:
  - 0 to the child process.
  - A positive non-zero value (the PID of the child) to the parent process.
- You can differentiate between the parent and child by checking the return value of fork(). If it returns 0, the process is the child; otherwise, it's the parent.