1. Tiến trình (Process) là một chương trình đang thực thi trên hệ thống.
2. Bố cục trong bộ nhớ của một tiến trình bao gồm các phần: Code, Data, Stack, Heap.
3. IPC là viết tắt của Inter-Process Communication, tức là giao tiếp giữa các tiến trình.
4. Để tạo tiến trình, hệ thống UNIX sử dụng lời gọi fork().
5. Liên kết mà 2 tiến trình sử dụng để gửi/nhận các thông điệp được gọi là kênh (channel).
6. Đường ống (pipe) là một cơ chế IPC cho phép truyền dữ liệu giữa các tiến trình thông qua một "ống".
7. Phương pháp dùng để thiết lập IPC bao gồm: Shared Memory, Message Passing, Synchronization Objects.
8. Một chương trình trở thành một tiến trình khi được thực thi trên hệ thống.
9. Để khởi chạy một chương trình trong Ubuntu, có thể sử dụng lệnh "./tên\_chương\_trình".
10. Một bộ IPC phải cung cấp tối thiểu những thao tác như tạo, gửi, nhận, xóa các thông điệp.
11. Hệ điều hành thực hiện chuyển ngữ cảnh (Context switch) khi tiến trình hiện tại phải dừng lại và cho phép tiến trình khác thực thi.
12. Giao tiếp trực tiếp (direct communication) là cơ chế IPC mà tiến trình gửi/nhận trực tiếp thông điệp với tiến trình khác.
13. Để cho 2 tiến trình giao tiếp gián tiếp (indirect communication), cần có một bộ trung gian như kênh (channel) hoặc bộ nhớ chung (shared memory).
14. Kỹ thuật gửi không-chặn giữa các tiến trình là kỹ thuật cho phép tiến trình gửi tin nhắn cho tiến trình khác mà không cần chờ đợi đến khi tiến trình kia đã nhận tin nhắn trước.
15. Khi sử dụng hàng chờ kích thước Zero, các phần tử sẽ được thêm vào hàng đợi chỉ khi có một tiến trình sẵn sàng nhận chúng.
16. Hàng chờ kích thước Zero được sử dụng trong giao tiếp giữa các tiến trình thông qua kỹ thuật gửi không-chặn.
17. Dung lượng có hạn hoặc dung lượng vô hạn là một tính chất của bộ nhớ.
18. Khi một tiến trình con được sinh ra bởi tiến trình cha, nó thừa kế các giá trị của thanh ghi và các biến được khởi tạo trước đó bởi tiến trình cha.
19. Nhiệm vụ của PCB là lưu trữ thông tin về tiến trình, bao gồm trạng thái tiến trình, thanh ghi, các biến và các thông tin khác.
20. Một tiến trình mang trạng thái "Ready" có thể chuyển sang trạng thái "Running".
21. Bước chuyển trạng thái "Running" sang trạng thái "Ready" là không tồn tại.
22. Hệ điều hành cần lưu trữ trạng thái của tiến trình B trước khi chuyển CPU đến tiến trình A.
23. PCB của một tiến trình bao gồm các thông tin về trạng thái tiến trình, thanh ghi, địa chỉ bộ nhớ, các biến và các thông tin khác.
24. Các hàng đợi dành cho các tiến trình thường được xây dựng bằng các cấu trúc dữ liệu như hàng đợi FIFO (First In First Out) hoặc hàng đợi ưu tiên.
25. Hai hình thức giao tiếp phổ biến giữa server và client là socket và RPC (Remote Procedure Call).
26. Trong UNIX, đường ống loại unnamed pipe được thiết kế cho giao tiếp giữa các tiến trình có quan hệ cha-con.
27. Khi thực thi đoạn mã, sẽ có hai tiến trình được tạo ra, bao gồm tiến trình cha và tiến trình con.
28. Mục tiêu của chuyển ngữ cảnh (Context Switch) là để chuyển CPU từ việc thực thi của một tiến trình sang tiến trình khác. Hạn chế của chuyển ngữ cảnh là tốn thời gian và tài nguyên của hệ thống.
29. Có 2 tiến trình được tạo ra.
30. Trong mã nguồn của tiến trình con, phát biểu đúng là "pid = fork();". Đoạn mã này dùng để tạo ra một tiến trình con mới.
31. Phát biểu đúng là “Mã nguồn của tiến trình con được viết bằng một ngôn ngữ lập trình cụ thể”.
32. Cơ chế "đường ống" (Pipe) là một cách để truyền dữ liệu giữa các tiến trình trong IPC (Inter-Process Communication). Nó được sử dụng để kết nối đầu ra của một tiến trình với đầu vào của một tiến trình khác, giúp truyền dữ liệu từ tiến trình này sang tiến trình kia. Pipe có thể được sử dụng trong cả trường hợp các tiến trình có quan hệ cha-con và các tiến trình độc lập.