

Bài 1. Lập trình đa tuyến. Yêu cầu:

Xét hai ma trận A và B trong đó A là ma trận M hàng và K cột; và B là ma trận K hàng và N cột. Tích của hai ma trận A và B là ma trận C có M hàng và N cột trong đó giá trị phần tử tại hàng i cột j là $C_{i,j}$ được tính theo công thức:

$$C_{i,j} = \sum_{n=1}^K A_{i,n} * B_{n,j}$$

Ví dụ, nếu A có 3 hàng 2 cột và B có 2 hàng 3 cột, thì phần tử $C_{3,1}$ được tính là tổng của $A_{3,1} * B_{1,1}$ và $A_{3,2} * B_{2,1}$.

Yêu cầu

- Bạn phải tạo ra $M \times N$ tuyến để tính toán. Mỗi tuyến tính toán một phần tử $C_{i,j}$ của ma trận kết quả.
- Các ma trận A và B phải được đọc từ file `data.dat` trong đó các số nguyên dương M , K và N được ghi ở dòng đầu tiên; M dòng sau là các hàng của ma trận A ; và K dòng tiếp theo ghi các hàng của ma trận B .

Gợi ý

- Tuyến cha sẽ tạo ra $M \times N$ tuyến con để tính toán, các tuyến này sẽ nhận tham số đầu vào là dòng i và cột j , tính toán phần tử $C_{i,j}$ của ma trận kết quả.
- Khi tất cả các tuyến đã tính toán xong, tuyến cha sẽ in ra màn hình ma trận kết quả C . Để làm điều này, tuyến cha phải đợi mọi tuyến tính toán kết thúc trước khi in ra kết quả.

Bài 2. Socket programming. Yêu cầu:

- Viết ứng dụng Socket phía Client
- Viết ứng dụng Socket phía Server
- Cho phép nhiều client có thể cùng giao tiếp với Server

Bài 3. Lập lịch tiến trình. Yêu cầu:

Cho dãy tiến trình p_1, p_2, \dots, p_n trong file *tientrinh.txt* trong đó dòng đầu tiên ghi số tiến trình, các dòng tiếp theo mỗi dòng ghi tên tiến trình và thời gian thực thi của tiến trình theo thứ tự đến của tiến trình trong hàng đợi.

Yêu cầu: Sử dụng thuật toán SJF có ưu tiên đưa ra danh sách thứ tự thực hiện của các tiến trình và thời gian chờ trung bình (average waiting time) của các tiến trình.

Chuẩn bị cho buổi báo cáo

- Sinh viên chuẩn bị ở nhà, đến buổi báo cáo sẽ bốc thăm chọn bài. Bốc thăm được bài nào sẽ báo cáo bài đó.
- Sau khi báo cáo xong, sinh viên trả lời 2-3 câu hỏi vấn đáp để thể hiện kiến thức, kỹ năng về các nội dung đã học trong chương trình.