

Lab 08: Contiguous Memory Allocation

Course: Operating Systems

Exercise 1

	First-Fit	Best-Fit	Worst-Fit
115	300	125	750
500	600	600	600
358	750	750	X
200	350	200	350
375	X	X	X

Nhận xét:

- + First-Fit sử dụng bộ nhớ chưa tối ưu, xảy ra hiện tượng phân mảnh nội khi các process yêu cầu vùng nhớ ít lại sử dụng phân vùng có kích thước lớn hơn, cụ thể là 115, sử dụng 300; 358 sử dụng 700; 200 sử dụng 350.
- + Best-Fit sử dụng bộ nhớ tối ưu hơn First-Fit, tránh được hiện tượng phân mảnh nội, tìm các phân vùng đủ với các yêu cầu vùng nhớ, chẳng hạn 115 chỉ dùng 125; 200 sử dụng vừa đủ 200.
- + Tuy nhiên, so về tốc độ cấp phát phân vùng, thuật toán Best-Fit tìm kiếm chậm hơn so với First-Fit khi duyệt toàn bộ các phân vùng trong bộ nhớ để tìm phân vùng thích hợp nhất. Do đó cần tổ chức thuật toán để tối ưu khả năng tìm kiếm cho Best-Fit.
- + Còn thuật toán Worst-Fit sử dụng vùng nhớ lãng phí, gây ra hiện tượng phân mảnh nội, đồng thời không đủ phân vùng trống cho các process yêu cầu vùng nhớ lớn hơn.

Exercise 2

	First-Fit	Best-Fit	Worst-Fit
Advantages	1. Cấp phát nhanh, thời gian yêu cầu thực thi ngắn 2. Hiện thực đơn giản	1. Cấp phát vùng nhớ tối ưu, tránh hiện tượng phân mảnh nội, giảm thiểu lãng phí vùng nhớ	Nhanh, đơn giản và dễ hiện thực
Disadvantages	1. Sử dụng lãng phí vùng nhớ, không cấp phát vùng nhớ tối ưu, gây ra hiện tượng phân mảnh nội 2. Yêu cầu tổ chức phân vùng hợp lí	1. Tìm kiếm không nhanh, yêu cầu thời gian tìm kiếm phân vùng thích hợp. 2. Hiện thực phức tạp, cần tổ chức thuật toán, phân vùng thích hợp để tối ưu tốc độ tìm kiếm	Lãng phí vùng nhớ, tiêu tốn thời gian, yêu cầu tìm kiếm, không tối ưu, gây hiện tượng phân mảnh nội.

■ ■ ■