

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
Khoa học - Kỹ thuật Máy tính



# HỆ ĐIỀU HÀNH

---

Bài thực hành số 8

## Contiguous Memory Allocation

---

GVHD: Hoàng Lê Hải Thanh

SV thực hiện: Trương Hoài Nam – 1813175

TP. HỒ CHÍ MINH, THÁNG 06/2020



## Mục lục

<b>1 EXERCISE</b>	<b>2</b>
1.1 Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? Rank the algorithms in terms of how efficiently they use memory.	2
1.2 Student write a short report that compares the advantages as well as disadvantages of the allocation algorithms, namely First-Fit, Best-Fit, Worst-Fit. . . . .	3

## 1 EXERCISE

1.1 Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? Rank the algorithms in terms of how efficiently they use memory.

**\* FIRST-FIT algorithm:**

- Process có kích thước **115 KB** được đặt vào khối: **300 KB**.
- Process có kích thước **500 KB** được đặt vào khối: **600 KB**.
- Process có kích thước **358 KB** được đặt vào khối: **750 KB**.
- Process có kích thước **200 KB** được đặt vào khối: **350 KB**.
- Process có kích thước **375 KB** được đặt vào khối: **750 KB**. Vì khi đặt process có kích thước 358 KB vào khối 750 KB thì khối này vẫn còn phần thừa bộ nhớ là:  $750 - 358 = 392$  KB.

**\* BEST-FIT algorithm:**

- Process có kích thước **115 KB** được đặt vào khối: **125 KB**.
- Process có kích thước **500 KB** được đặt vào khối: **600 KB**.
- Process có kích thước **358 KB** được đặt vào khối: **750 KB**.
- Process có kích thước **200 KB** được đặt vào khối: **200 KB**.
- Process có kích thước **375 KB** được đặt vào khối: **750 KB**. Vì khi đặt process có kích thước 358 KB vào khối 750 KB thì khối này vẫn còn phần thừa bộ nhớ là:  $750 - 358 = 392$  KB.

**\* WORST-FIT algorithm:**

- Process có kích thước **115 KB** được đặt vào khối: **750 KB**.
- Process có kích thước **500 KB** được đặt vào khối: **600 KB**.
- Process có kích thước **358 KB** được đặt vào khối: **750 KB**. Vì khi đặt process có kích thước 115 KB vào khối 750 KB thì khối này vẫn còn phần thừa bộ nhớ là:  $750 - 115 = 635$  KB.
- Process có kích thước **200 KB** được đặt vào khối: **200 KB**.
- Process có kích thước **375 KB** sẽ phải đợi các process khác hoàn thành để được nạp vào memory. Vì process có kích thước 375 KB đã lớn hơn kích thước các memory đang có hiện tại (Khối 300 KB, Khối 600 KB hiện tại còn 100 KB, Khối 350 KB, Khối 200 KB hiện tại còn 0 KB, Khối 750 KB hiện tại còn 277 KB, Khối 125 KB).

⇒ Dựa vào kết quả có như trên, có thể xếp hạng giải thuật theo thứ tự sử dụng hiệu quả bộ nhớ (memory) giảm dần lần lượt là: Best-fit » First-fit » Worst-fit.

## 1.2 Student write a short report that compares the advantages as well as disadvantages of the allocation algorithms, namely First-Fit, Best-Fit, Worst-Fit.

### \* FIRST-FIT algorithm:

- Ưu điểm:

- Là giải thuật nhanh nhất vì nó tìm kiếm ít nhất.

- Nhược điểm:

- Lãng phí nhiều bộ nhớ, có thể có vùng nhớ còn thừa sau khi được cấp phát cho process.
- Giải thuật chọn vùng nhớ thích hợp kể từ đầu bộ nhớ, do đó nếu process kích thước lớn nhưng đến sau có thể không được cấp phát.

### \* BEST-FIT algorithm:

- Ưu điểm:

- Sử dụng bộ nhớ hiệu quả, do hệ điều hành phân bổ không gian công việc tối thiểu có thể có trong bộ nhớ, tránh được việc lãng phí, ngoài ra còn quản lý bộ nhớ hiệu quả hơn.

- Nhược điểm:

- Giải thuật khá chậm vì mỗi lần muốn cấp phát vùng nhớ thích hợp phải tìm kiếm toàn bộ bộ nhớ.
- Vì chọn vùng nhớ trống thích hợp nhất và nhỏ nhất nên khả năng gây ra phân mảnh ngoại cao.

### \* WORST-FIT algorithm:

- Ưu điểm:

- Tránh hiện tượng tạo ra các khối nhớ trống rất nhỏ.

- Nhược điểm:

- Giải thuật khá chậm vì mỗi lần muốn cấp phát vùng nhớ lớn nhất phải tìm kiếm toàn bộ bộ nhớ.
- Các process rất lớn nếu đến sau có thể không được cấp phát, vì các vùng nhớ trống còn lại không đủ kích thước.