

Bài tập Hệ Điều Hành

bài Tập Thuật Giải Round - Robin tham Khảo!!

Bài tập Thuật giải Round - Robin tham khảo!!

Một hệ thống có 3 tiến trình với thời điểm đến và thời gian sử dụng CPU như sau:

TT Thời điểm đến (ms) CPU-Burst (ms)

P1 4 46

P2 30 28

P3 51 33

Dùng thuật giải RRS với thời lượng bằng 20 ms để điều phối CPU (có thể có 2 phương án):

a. Thể hiện bằng biểu đồ Gantt.

b. Tính thời gian chờ trung bình của các tiến trình.

Mình giải kết quả như sau mong các bạn đóng góp ý kiến.

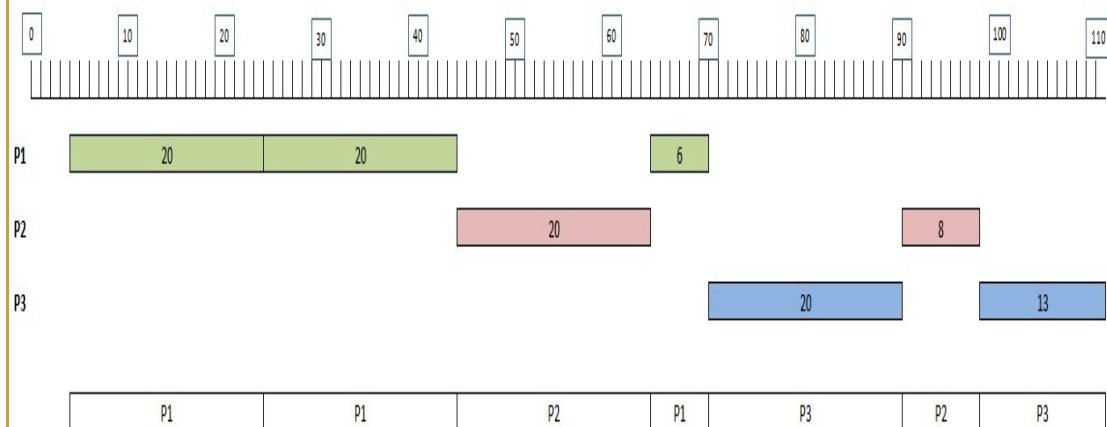
a/

```
-----+----P1--+----P1----+----P2-----+P1--+----P3-----+--P2--+-- P3-----  
-----|-----|-----|-----|-----|-----|-----|-----|  
0 --- 4 -----24 ----- 44 ----- 64 --- 70 ----- 90 ---- 98 -----111
```

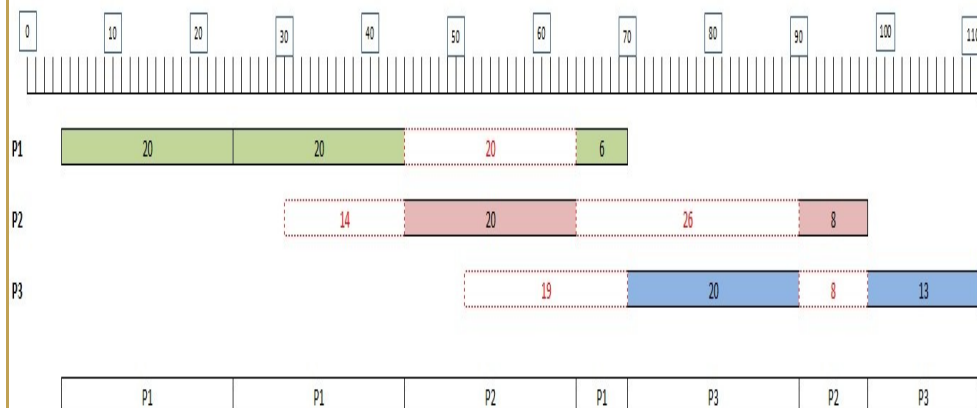
b/ TG chờ TB = $87/3=29$ (ms)

Giải thích:

Hình vẽ mô tả các tiến trình đây:



1. Thời điểm 4: P1 bắt đầu chạy 20 ms
2. Thời điểm 24: do P2,P3 chưa đến nên P1 chạy tiếp 20 ms
3. Thời điểm 44: P2 được tiếp quyền P1 (vì P2 đang chờ ở thời điểm 30)
4. Thời điểm 64: P1 được tiếp quyền P2 (vì P1 đang chờ ở thời điểm 44)
5. Thời điểm 70: P3 được chạy trước P2 (vì P3 đang chờ trước ở thời điểm 51 còn P2 là 64)
6. Thời điểm 90: P2 được tiếp quyền P3 (vì P2 đang chờ ở thời điểm 64)
7. Cuối cùng chỉ còn P3 sẽ chạy hết thời gian còn lại.



Chú ý thời điểm đến của P2=30 và P3=51

Bạn cộng các số màu đỏ trong hình lại rồi chia cho 3.

Kết quả : $87/3=29$.

bài Tập Quản lý bộ Nhớ chính (Kiểm tra Lỗi khung trang)

1./ Xét chuỗi truy xuất bộ nhớ sau:

1,2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3

Giả sử bộ nhớ vật lý có 4 khung trang. Minh họa kết quả trình thay thế trang với các thuật toán thay thế sau:

a)FIFO

b) OPT

c) LRU

Giải

a) FIFO

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3
*	*	*	*			*	*	*	*		*	*	*	
①	1	1	1	1	1	⑤	5	5	5	5	③	3	3	3
	②	2	2	2	2	2	⑥	6	6	6	6	⑦	7	7
		③	3	3	3	3	3	②	2	2	2	2	⑥	6
			④	4	4	4	4	4	①	1	1	1	1	1

→ 11 lỗi trang

b) OPT

+	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3
	*	*	*	*			*	*					*		
①	1	1	1	1	1	1	1	1	1	1	1	1	⑦	7	7
	②	2	2	2	2	2	2	2	2	2	2	2	2	2	2
		③	3	3	3	3	3	3	3	3	3	3	3	3	3
			④	4	4	⑤	⑥	6	6	6	6	6	6	6	6

→ 7 lỗi trang

c) LRU

1	2	3	4	2	1	5	6	2	1	2	3	7	6	3
*	*	*	*			*	*				*	*	*	
①.	1	1	1	1	1	1	1	1	1	1	1	1	⑥.	6
	②.	2	2	2	2	2	2	2	2	2	2	2	2	2
		③.	3	3	3	⑤.	5	5	5	5	③.	3	3	3
			④.	4	4	4	⑥.	6	6	6	6	⑦.	7	7

Dấu chấm để đánh dấu truy cập. Lâu nhất chưa truy cập

→ 9 lỗi trang.

Chú ý:

- Thuật toán FIFO: Trong các trang đang ở trong bộ nhớ, chọn trang được nạp vào bộ nhớ trước nhất để thay thế.
- Thuật toán OPT: Chọn trang sẽ lâu được sử dụng nhất trong tương lai để thay thế.
- Thuật toán LRU: Chọn trang lâu nhất chưa được sử dụng.

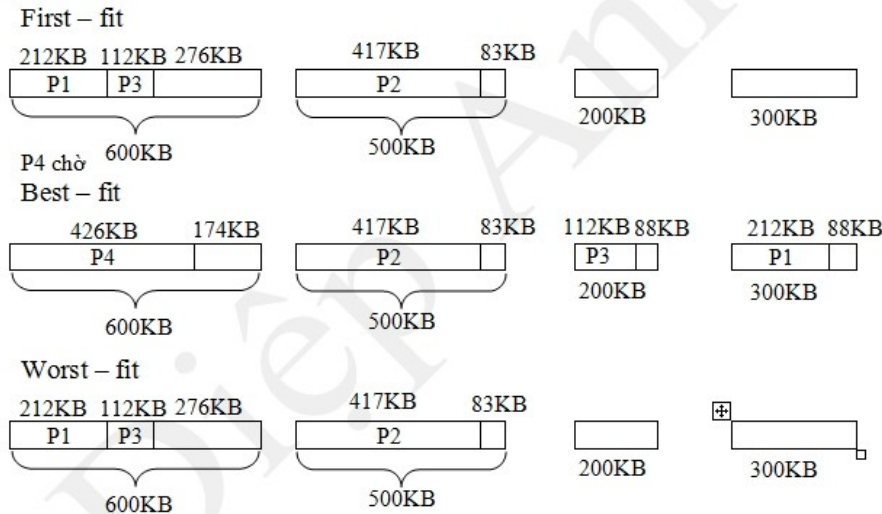
bài Tập Cấp phát bộ Nhớ (First-fit, Best-first, Worst-fit)

Best-first

1./ Trong mô hình cấp phát bộ nhớ liên tục, có năm phân mảnh bộ nhớ theo thứ

tự với kích thước là 600KB, 500KB, 200KB, 300KB. Giả sử có 4 tiến trình đang chờ cấp phát bộ nhớ theo thứ tự P1, P2, P3, P4. Kích thước tương ứng của các tiến trình trên là: 212KB, 417KB, 112KB, 426KB. Hãy cấp phát bộ nhớ cho các tiến trình trên theo thuật toán First-fit, Best-first, Worst-fit.

Giải

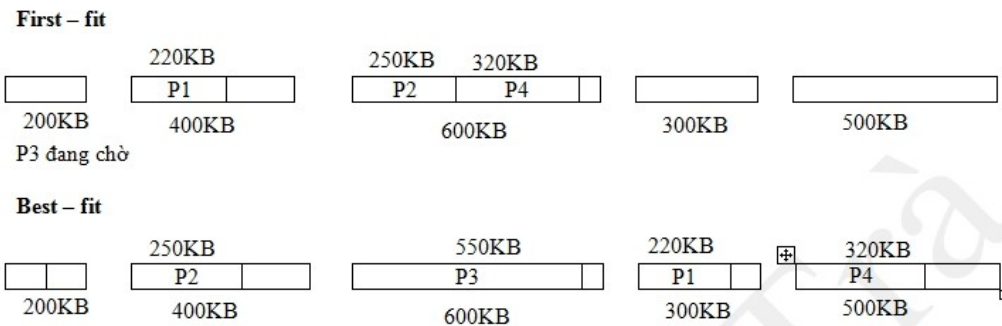


P4 chờ

2./ (đề kiểm tra) Trong mô hình cấp phát bộ nhớ liên tục, có 5 phân mảnh bộ nhớ với kích thước là 200KB, 400KB, 600KB, 300KB, 500KB. Giả sử có 4 tiến trình đang chờ cấp phát bộ nhớ theo thứ tự P1, P2, P3, P4. Kích thước tương ứng các tiến trình trên là: 220KB, 250KB, 550KB, 320KB.

Hãy cấp phát bộ nhớ cho các tiến trình trên theo thuật toán First – fit và Best – fit.

Giải



Chú ý: - First – fit : tìm vùng nhớ đầu tiên đủ lớn để chứa tiến trình

- Best – fit: tìm vùng nhớ nhỏ nhất mà có thể chứa tiến trình

- Worst – fit: tìm vùng nhớ lớn nhất cấp cho tiến trình.

bài Tập Quản lý Tiến trình Bằng các Chiến Lược Điều phối FIFO, SJF, RR,

1./ Xét tập hợp các tiến trình sau:

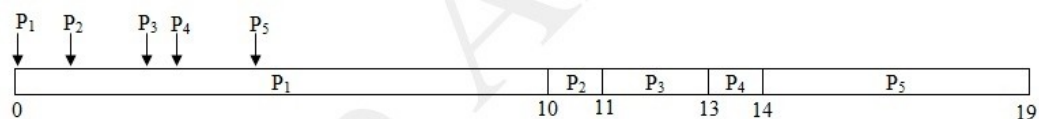
Tiến trình	Thời điểm vào RL	Thời gian CPU	Độ ưu tiên
P ₁	0	10	3
P ₂	1	1	1
P ₃	2.5	2	3
P ₄	3	1	4
P ₅	4.5	5	2

Hãy cho biết kết quả điều phối theo các chiến lược

- FIFO
- SJF
- Round Robin với $q = 2$
- Độ ưu tiên độc quyền
- Độ ưu tiên không độc quyền
- tính thời gian chờ cho từng tiến trình và thời gian chờ trung bình trong các chiến lược trên.

Giải

a./ FIFO



Thời gian chờ:

P1: 0

P2: $10 - 1 = 9$

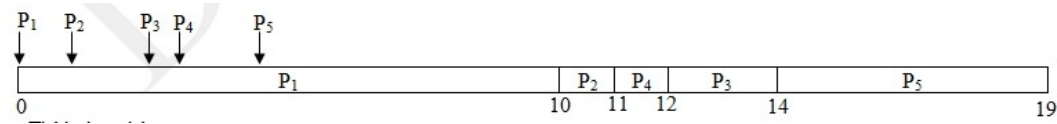
P3: $11 - 2.5 = 8.5$

P4: $13 - 3 = 10$

P5: $14 - 4.5 = 9.5$

Thời gian chờ trung bình = $37/5 = 7.45$

b./ SJF



Thời gian chờ:

P1: 0

P2: $10 - 1 = 9$

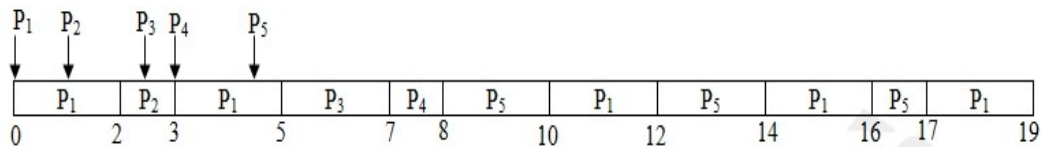
P3: $12 - 2.5 = 9.5$

P4: $11 - 3 = 8$

P5: $14 - 4.5 = 9.5$

Thời gian chờ trung bình = $36/5 = 7.2$

c./ Round Robin



Thời gian chờ:

P1: $1 + 5 + 2 + 1 = 9$

P2: $2 - 1 = 1$

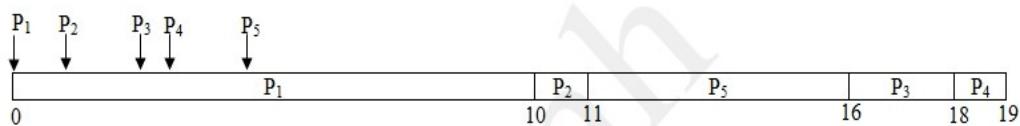
P3: $5 - 2.5 = 2.5$

P4: $7 - 3 = 4$

P5: $8 + 2 + 2 - 4.5 = 7.5$

Thời gian chờ trung bình = $25/5 = 5$

d./ Độ ưu tiên độc quyền



Thời gian chờ:

P1: 0

P2: $10 - 9 = 1$

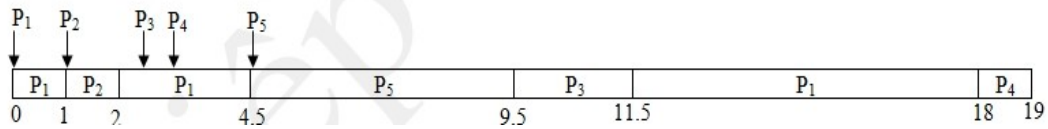
P3: $16 - 2.5 = 13.5$

P4: $18 - 3 = 5$

P5: $11 - 4.5 = 6.5$

Thời gian chờ trung bình = $44/5 = 8.8$

e./ Độ ưu tiên không độc quyền



Thời gian chờ:

P1: $1 + 7 = 8$

P2: 0

P3: $9.5 - 2.5 = 7$

P4: $18 - 3 = 15$

P5: 0

Thời gian chờ trung bình = $25/5 = 5$

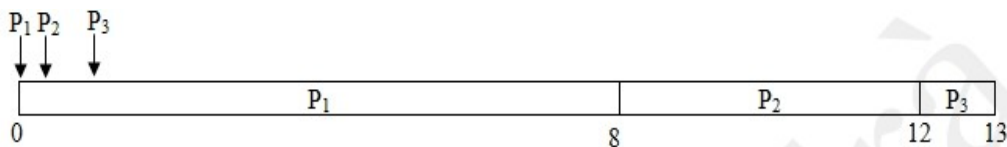
2./ Cho các tiến trình sau:

Tiến trình	Thời điểm vào RL	Thời gian CPU
P ₁	0	8
P ₂	0.4	4
P ₃	1	1

Hãy cho biết các kết quả điều phối chiến lược FIFO và SJF và thời gian chờ của từng chiến lược

Giải

a./ FIFO



Thời gian chờ

P1: 0

P2: $8 - 0.4 = 7.6$

P3: $12 - 1 = 11$

Thời gian chờ trung bình = $18.6 / 3 = 6.2$

b./ SJF



P₁: 0

P₂: $9 - 0.4 = 8.6$

P₃: $8 - 1 = 7$

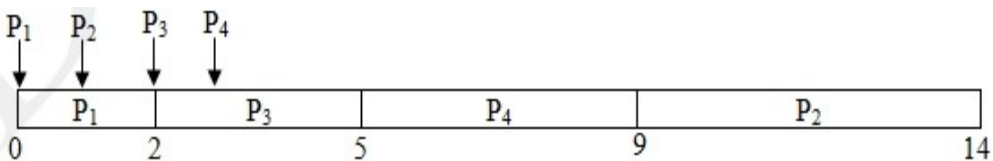
Thời gian chờ trung bình = $15.6 / 3 = 5.2$

3./ Điều phối các tiến trình sau theo chiến lược điều phối độ ưu tiên độ quyền.

Tiến trình	Chiều dài CPU burst	Thời điểm vào RL	Độ ưu tiên
P ₁	2	0	2
P ₂	5	1	3
P ₃	3	2	1
P ₄	4	3	0

Tính thời gian chờ cho từng tiến trình và thời gian chờ trung bình.

Giải



Thời gian chờ:

P₁: 0

P₂: $9 - 1 = 8$

P₃: 0

P₄: $5 - 3 = 2$

Thời gian chờ trung bình = $10/4 = 2.5$

Chú ý:

- FIFO vào trước thực hiện trước.
- SJF tiến trình nào có chiều dài CPU burst ngắn thì thực hiện trước.
- RR mỗi tiến trình chỉ được thực hiện trong một thời gian q nhất định, các tiến

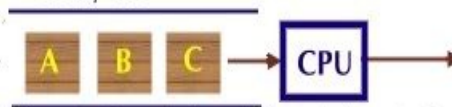
trình lần lượt thực hiện xoay vòng.

- Điều phối theo độ ưu tiên độc quyền: có độ ưu tiên nhỏ thực hiện trước.
- Điều phối ưu tiên không độc quyền: giống như trên nhưng nếu đang thực hiện mà xuất hiện tiến trình có độ ưu tiên nhỏ hơn thì phải dừng để nhường cho tiến trình kia thực hiện.

các Chiến Lược Điều Phối Hệ Điều hành

1. Điều phối FIFO

CPU được cấp phát cho tiến trình đầu tiên trong danh sách sẵn sàng có yêu cầu, là tiến trình được đưa vào hệ thống sớm nhất. Đây là thuật toán điều phối theo nguyên tắc độc quyền. Một khi CPU được cấp phát cho tiến trình, CPU chỉ được tiến trình tự nguyện giải phóng khi kết thúc xử lý hay khi có một yêu cầu nhập/xuất.



Hình 2.12 Điều phối FIFO

► Ví dụ :

Tiến trình	Thời điểm vào RL	Thời gian xử lý
P1	0	24
P2	1	3
P3	2	3

Thứ tự cấp phát CPU cho các tiến trình là :

P1	P2	P3
0	24	27 30

*Thời gian xử lý : $P1=24, P2=3, P3=3$

* Thời gian xử lý trung bình : $(24+3+3)/3=10$

*Thời gian đợi : $P1=0$; $P2=24-1 = 23$; $P3 = 24+3-2 = 25$

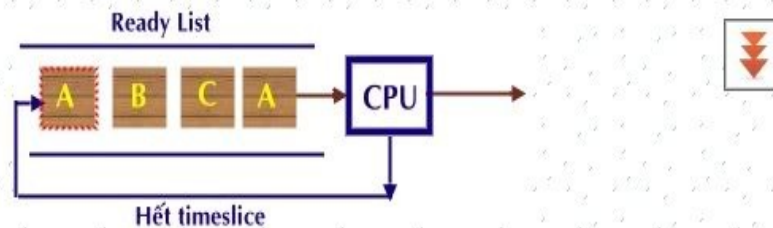
*Thời gian đợi trung bình: $(0+23+25)/3 = 16$

* Thời gian lưu lại trong hệ thống : $P1=24$; $P2=3$; $P3=3$

* Thời gian lưu lại trung bình : $(24+3+3)/3 = 10$

2.Chiến lược điều phối xoay vòng :

Danh sách sẵn sàng được xử lý như một danh sách vòng, bộ điều phối lần lượt cấp phát cho từng tiến trình trong danh sách một khoảng thời gian sử dụng CPU gọi là [i]quantum. Đây là một giải thuật điều phối không độc quyền : khi một tiến trình sử dụng CPU đến hết thời gian quantum dành cho nó, hệ điều hành thu hồi CPU và cấp cho tiến trình kế tiếp trong danh sách. Nếu tiến trình bị khóa hay kết thúc trước khi sử dụng hết thời gian quantum, hệ điều hành cũng lập tức cấp phát CPU cho tiến trình khác. Khi tiến trình tiêu thụ hết thời gian CPU dành cho nó mà chưa hoàn tất, tiến trình được đưa trở lại vào cuối danh sách sẵn sàng để đợi được cấp CPU trong lượt kế tiếp.



Hình 2.13 Điều phối Round Robin

Tiến trình	Thời điểm vào RL	Thời gian xử lý
P1	0	24
P2	1	3
P3	2	3

Nếu sử dụng quantum là 4 miliseconds, thứ tự cấp phát CPU sẽ là :

P1	P2	P3	P1	P1	P1	P1	P1
0	4	7	10	14	18	22	26 30

*Thời gian xử lý : $P1=24, P2=3, P3=3$

* Thời gian xử lý trung bình : $(24+3+3)/3=10$

*Thời gian đợi :

$P1=0$

$P2=4-1=3$

$P3=4+3-2=5$

$$P1' = 4 + 3 + 3 - 3 = 6$$

đến lúc này p1 sẽ được xử lý liên tục nên không phải tính thời gian chờ của p1'' p1'''...

* Thời gian đợi trung bình: $(0 + 3 + 5 + 6) / 3 = 4.66$

* Thời gian lưu lại trong hệ thống :

P1: vòng 1 = 4 vòng 2 = 20 khoảng cách 2 vòng là 6 => thời gian lưu lại của P1 = $4 + 20 + 6 = 30$

$$P2 = 3$$

$$P3 = 3$$

* Thời gian lưu lại trung bình : $(30 + 3 + 3) / 3 = 12$

.Điều phối với độ ưu tiên

Mỗi tiến trình được gán cho một độ ưu tiên tương ứng, **tiến trình có độ ưu tiên cao nhất sẽ được chọn để cấp phát CPU đầu tiên**. Độ ưu tiên có thể được định nghĩa nội tại hay nhờ vào các yếu tố bên ngoài. Độ ưu tiên nội tại sử dụng các đại lượng có thể đo lường để tính toán độ ưu tiên của tiến trình, ví dụ các giới hạn thời gian, nhu cầu bộ nhớ... Độ ưu tiên cũng có thể được gán từ bên ngoài dựa vào các tiêu chuẩn do hệ điều hành như tầm quan trọng của tiến trình, loại người sử dụng sở hữu tiến trình... Giải thuật điều phối với độ ưu tiên có thể theo nguyên tắc **độc quyền hay không độc quyền**. Khi một tiến trình được đưa vào danh sách các tiến trình sẵn sàng, độ ưu tiên của nó được so sánh với độ ưu tiên của tiến trình hiện hành đang xử lý. Giải thuật điều phối **với độ ưu tiên và không độc quyền sẽ thu hồi CPU từ tiến trình hiện hành để cấp phát cho tiến trình mới** nếu độ ưu tiên của tiến trình này cao hơn tiến trình hiện hành. Một giải thuật **độc quyền sẽ chỉ đơn giản chen tiến trình mới vào danh sách sẵn sàng, và tiến trình hiện hành vẫn tiếp tục xử lý hết thời gian dành cho nó**.

► Ví dụ: (độ ưu tiên 1 > độ ưu tiên 2 > độ ưu tiên 3)

Tiến trình	Thời điểm vào RL	Độ ưu tiên	Thời gian xử lý
P1	0	3	24
P2	1	1	3
P3	2	2	3

Sử dụng thuật giải độc quyền, thứ tự cấp phát CPU như sau :

P1	P2	P3
0	24	27 30

Sử dụng thuật giải không độc quyền, thứ tự cấp phát CPU như sau :

P1	P2	P3	P1
0	1	4	7 30

* Trường hợp độc quyền

* Thời gian xử lý : $P1=24, P2=3, P3=3$

* Thời gian xử lý trung bình : $(24+3+3)/3=10$

* Thời gian đợi : $P1=0; P2=23; P3=25$

* Thời gian đợi trung bình: $(0+23+25)/3=16$

* Thời gian lưu lại trong hệ thống : $P1=24; P2=3; P3=3$

* Thời gian lưu lại trung bình: $(24+3+3)/3=10$

* Trường hợp không độc quyền:

* Thời gian xử lý : $P1=24, P2=3, P3=3$

* Thời gian xử lý trung bình : $(24+3+3)/3=10$

* Thời gian đợi :

$P1=0$

$P2=0+1-1=0$

$P3=0+1+3-2=2$

$P1'=0+1+3+3-3=4$

*Thời gian đợi trung bình: $(0+0+2+4)/3=2$

* Thời gian lưu lại trong hệ thống :

P1 Vòng 1 = 1 vòng 2= 23 khoảng cách giữa 2 vòng =6 =>
thời gian lưu lại của p1= $1+23+6=30$

p2=3

p3=3

* Thời gian lưu lại trung bình : $(30+3+3)/3=12$

4.Chiến lược công việc ngắn nhất (Shortest-job-first SJF)

Đây là một trường hợp đặc biệt của giải thuật điều phối với độ ưu tiên. Trong giải thuật này, độ ưu tiên p được gán cho mỗi tiến trình là nghịch đảo của thời gian xử lý mà tiến trình yêu cầu : $p = 1/t$. Khi CPU được tự do, nó sẽ được cấp phát cho tiến trình yêu cầu ít thời gian nhất để kết thúc- tiến trình ngắn nhất.

Giải thuật này cũng có thể **độc quyền** hay **không độc quyền**. Sự chọn lựa xảy ra khi có một tiến trình mới được đưa vào danh sách sẵn sàng trong khi một tiến trình khác đang xử lý. Tiến trình mới có thể sở hữu một yêu cầu thời gian sử dụng CPU cho lần tiếp theo (CPU-burst) ngắn hơn thời gian còn lại mà tiến trình hiện hành cần xử lý. Giải thuật SJF không độc quyền sẽ dừng hoạt động của tiến trình hiện hành, trong khi giải thuật độc quyền sẽ cho phép tiến trình hiện hành tiếp tục xử lý.

Tiến trình	Thời điểm vào RL	Thời gian xử lý
P1	0	6
P2	1	8
P3	2	4
P4	3	2

Sử dụng thuật giải SJF độc quyền, thứ tự cấp phát CPU như sau:

P1	P4	P3	P2
0	6	8	12,20

Sử dụng thuật giải SJF không độc quyền, thứ tự cấp phát CPU như sau:

P1	P4	P1	P3	P2
0	3	5	8	12,20

* Trường hợp độc quyền

- * Thời gian xử lý : $P1=6, P2=8, P3=4; P4=2$
- * Thời gian xử lý trung bình : $(6+8+4+2)/4=5$
- * Thời gian đợi : $P1=0$; $p4=5$; $p3=6$; $p2=9$
- * Thời gian đợi trung bình: $(0+5+6+9)/4=5$
- * Thời gian lưu lại trong hệ thống : $P1=6$; $P4=2$; $P3=4$; $P2=8$
- * Thời gian lưu lại trung bình: $(6+2+4+8)/4=5$

* Trường hợp không độc quyền:

- * Thời gian xử lý : $P1=6, P2=8, P3=4; P4=2$
- * Thời gian xử lý trung bình : $(6+8+4+2)/4=5$
- * Thời gian đợi :
- $P1=0$
- $P4=2$
- $P1'=3$
- $P3=5$
- $P2=8$
- * Thời gian đợi trung bình: $(0+2+3+5+8)/4=4.5$

* Thời gian lưu lại trong hệ thống :

P1 Vòng 1 = 3 vòng 2 = 3 khoảng cách giữa 2 vòng = 2 => thời gian lưu lại của p1 = $3+3+2=8$

P4=2

p3=4

P2=8

* Thời gian lưu lại trung bình : $(8+2+4+8)/4=5$