



BÀI 3

LẬP LỊCH



MỤC TIÊU

- Nắm được các kiến thức liên quan đến mục tiêu của việc lập lịch.
- Nắm được các thuật toán lập lịch, ưu nhược điểm của các thuật toán.
- Phân biệt sự khác nhau giữa các thuật toán lập lịch.
- Trình bày được về một thuật toán lập lịch.
- Bạn nên dành 5 tiết học lý thuyết, 3 tiết làm bài tập.



NỘI DUNG

- 1. Các khái niệm cơ bản**
- 2. Các tiêu chuẩn lập lịch**
- 3. Các thuật toán lập lịch và đánh giá thuật toán**



1. MỘT SỐ KHÁI NIỆM CƠ BẢN

1. MỘT SỐ KHÁI NIỆM CƠ BẢN

- **Đa chương trình (Mutilprogramming):** Chế độ ĐCT cho phép khi một chương trình đang được thực hiện ở thiết bị ngoại vi thì bộ xử lý trung tâm có thể được sử dụng cho các chương trình khác, tạo ra khả năng nhiều chương trình được thực hiện đồng thời trên một máy tính.
- **Chu kỳ CPU - I/O:** Thực thi chương trình gồm một chu kỳ thực thi CPU và chờ đợi nhập xuất, các quá trình này thì đan xen lẫn nhau.
- **Giới hạn CPU (CPU - Bound):** Tiến trình có thời gian sử dụng CPU nhiều hơn thời gian sử dụng I/O.
- **Giới hạn nhập xuất (I/O - bound):** Tiến trình dùng phần lớn thời gian để đợi I/O.

...

Load store
add store
read from file

CPU
Burst

Wait for I/O

I/O Burst

Store increment
index write to file

CPU
Burst

Wait for I/O

I/O Burst

Load store add
store read from file

CPU
Burst

Wait for I/O

I/O Burst

...

Thay đổi tuần tự của chu kỳ CPU-I/O

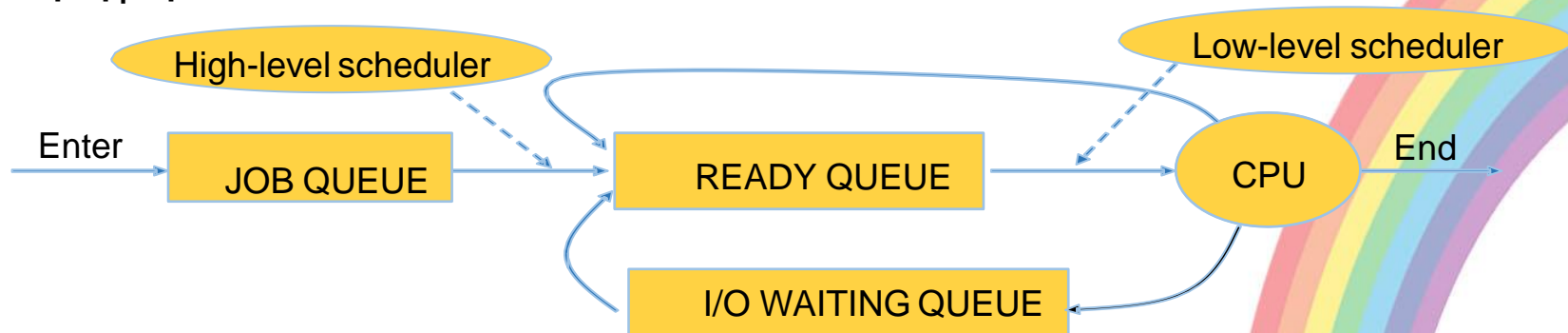


1. MỘT SỐ KHÁI NIỆM CƠ BẢN

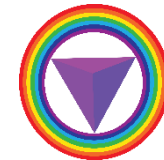


1.1. LẬP LỊCH CPU

- Khái niệm: Việc chọn một tiến trình đã sẵn sàng trong bộ nhớ để cho CPU thực hiện;
- Lập lịch cho CPU khi tiến trình:
 - 1) Chuyển từ Running -> Ready
 - 2) Chuyển từ Running -> Waiting
 - 3) Chuyển từ Waiting -> Ready
 - 4) Kết thúc:
 - Trường hợp 2 và 4: Lập lịch không ưu tiên (Nonpreemptive);
 - Trường hợp 1 và 3: Lập lịch ưu tiên (Preemptive).
- Bộ lập lịch CPU:



1. MỘT SỐ KHÁI NIỆM CƠ BẢN

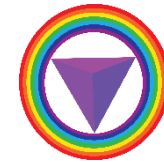


1.2. MỤC TIÊU ĐỊNH THỜI

- Công bằng;
- Tăng hiệu suất tối đa;
- Cực đại số người dùng tương tác;
- Có thể dự đoán trước;
- Phí tổn ít;
- Cân đối việc sử dụng tài nguyên;
- Tránh trì hoãn vô hạn định (dùng độ ưu tiên);
- Ưu tiên quá trình giữ tài nguyên quan trọng;
- Phục vụ tốt các quá trình có hướng thuận lợi;
- Điều phối tối ưu khi tải không cân đối.



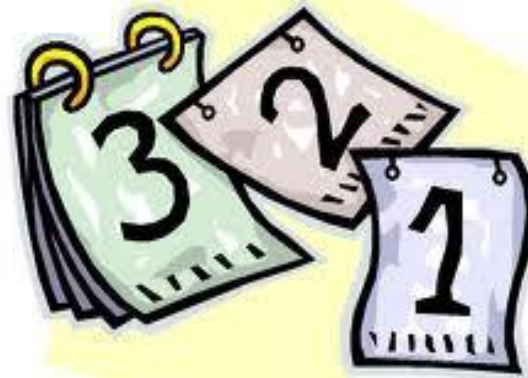
1. MỘT SỐ KHÁI NIỆM CƠ BẢN



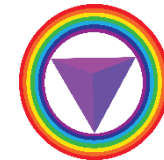
1.3. CÁC MỨC LẬP LỊCH

Có 3 mức:

- Lập lịch mức cao: Xác định một chương trình có được thực thi hay không?
- Lập lịch mức giữa: Xác định những tiến trình nào đã sẵn sàng.
- Lập lịch mức thấp: Xác định tiến trình tiếp theo sẽ được thực thi.



1. MỘT SỐ KHÁI NIỆM CƠ BẢN



1.4. BỘ PHÂN PHỐI (DISPATCHER)

- Chuyển điều khiển CPU sang cho quá trình;
- Thực hiện bước chuyển ngữ cảnh:
 - Chuyển ngữ cảnh sang cấp người dùng;
 - Chuyển chế độ người dùng;
 - Nhảy sang vị trí thích hợp của quá trình và thực thi.
- Thời gian chờ chuyển đổi (Dispatch Latency): Thời gian để bộ phân phối dừng trạng thái đang thực thi để chuyển sang trạng thái khác.





2. CÁC TIÊU CHUẨN LẬP LỊCH



2. CÁC TIÊU CHUẨN LẬP LỊCH

- Độ lợi CPU: CPU bận nhiều nhất có thể.
- Thông lượng: số tiến trình được thực hiện trong một đơn vị thời gian.
- Thời gian xử lý: Tổng thời gian một tiến trình được thực thi.
- Thời gian chờ: Tổng thời gian tiến trình phải chờ trong hàng đợi sẵn sàng.
- Thời gian đáp ứng: Tổng thời gian từ khi có yêu cầu đến khi có đáp ứng đầu tiên, không gồm thời gian xuất ra đáp ứng đó.





2. CÁC TIÊU CHUẨN LẬP LỊCH

CÁC TIÊU CHUẨN TỐI ƯU

- Độ lợi CPU là lớn nhất;
- Thông lượng lớn nhất;
- Thời gian xử lý nhỏ nhất;
- Thời gian chờ nhỏ nhất;
- Thời gian đáp ứng nhỏ nhất.





3. CÁC THUẬT TOÁN LẬP LỊCH



3. CÁC THUẬT TOÁN LẬP LỊCH

- Theo thời gian kết thúc (Deadline);
- FIFO;
- Round-Robin (RR);
- Shortest Job First (SJF);
- Shortest Rest Time (SRT);
- Highest Response Ratio Next;
- Hàng đợi nhiều mức với mỗi liên hệ ngược (Call Back).





3. CÁC THUẬT TOÁN LẬP LỊCH

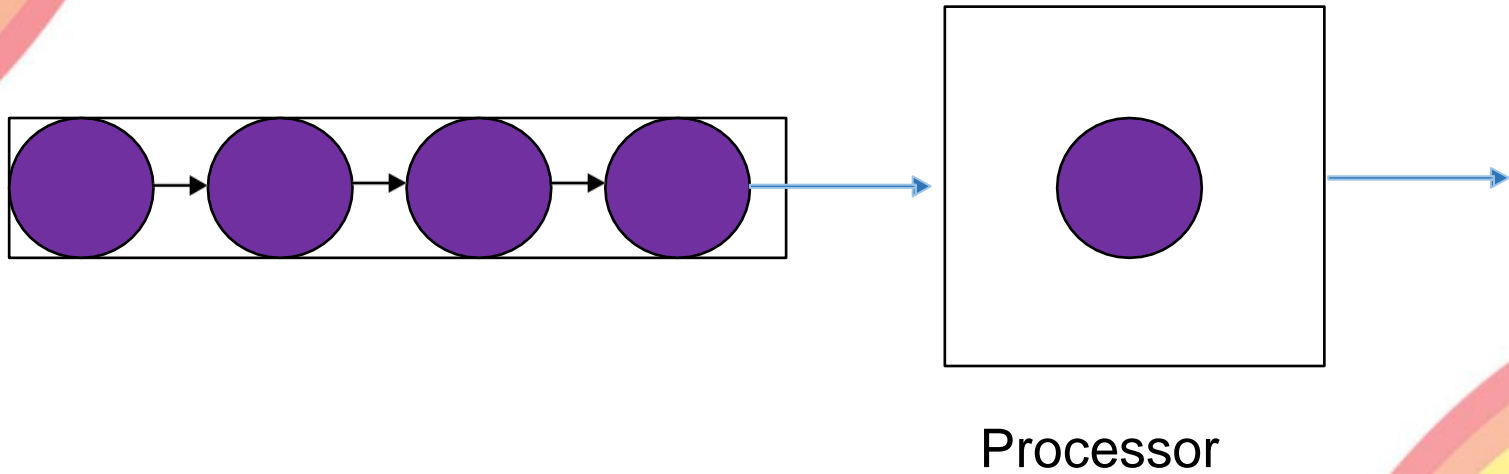
3.1. LẬP LỊCH THEO THỜI GIAN KẾT THÚC

- Lập lịch sao cho các tiến trình kết thúc trong thời hạn định trước.
- Nhược điểm:
 - Khó khăn trong việc chỉ rõ tài nguyên cần thiết phục vụ cho process;
 - Vừa phải thực hiện ứng dụng đúng hạn, vừa không làm ảnh hưởng “ quá nhiều” đến các ứng dụng khác;
 - Tranh chấp tài nguyên giữa các ứng dụng;
 - Phức tạp khi có nhiều yêu cầu kết thúc các ứng dụng đúng thời hạn;
 - Chi phí tài nguyên có thể lớn hơn, làm ảnh hưởng đến cả hệ thống.



3. CÁC THUẬT TOÁN LẬP LỊCH

3.2. GIẢI THUẬT FIFO



- FIFO: First in First out hay còn gọi là **FCFS (*First Come First Served*)**;
- Lựa chọn tiến trình: Đợi trong hàng đợi ready lâu nhất;
- Định thời theo kiểu không hoán đổi.



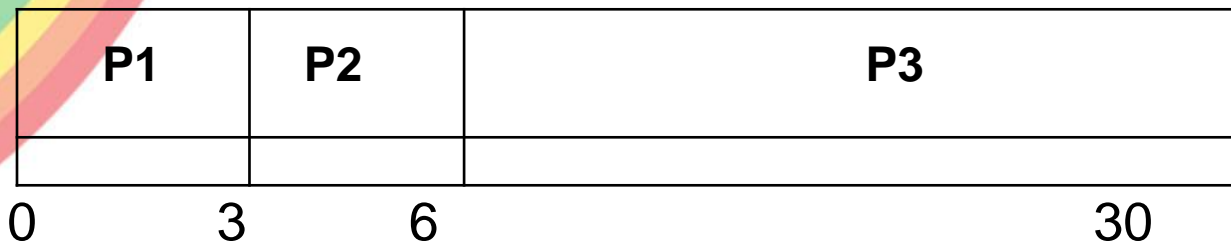
- Giả sử các tiến trình đến theo thứ tự P1, P2, P3;
- Giản đồ Gantt** cho việc định thời là:

- Công thức tính thời gian đợi = Thời điểm thực thi - Thời điểm xuất hiện
Thời gian đợi cho P1 = 0 – 0 = 0; P2 = 24 – 1 = 23; P3 = 27 – 2 = 25.
- Thời gian đợi trung bình: $(0 + 23 + 25)/3 = 16$.



VÍ DỤ 2

Giả sử các tiến trình đến theo thứ tự: P1, P2, P3. Giải đồ Gantt cho việc định thời là:

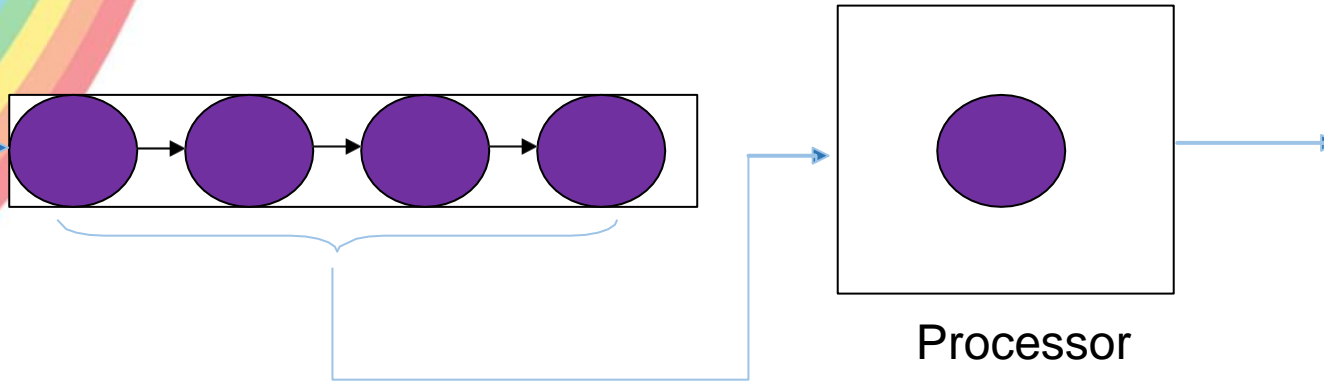


- Thời gian đợi cho P1 = 0, P2 = 2, P3 = 4;
- Thời gian đợi trung bình là: $(0 + 2 + 4)/3 = 2$;
- Nhận xét:
 - Tốt hơn rất nhiều so với trường hợp trước;
 - Hiệu quả hơn khi xử lý tiến trình ngắn trước tiến trình dài.



3. CÁC THUẬT TOÁN LẬP LỊCH

3.3. GIẢI THUẬT SJF (Shortest Job First)



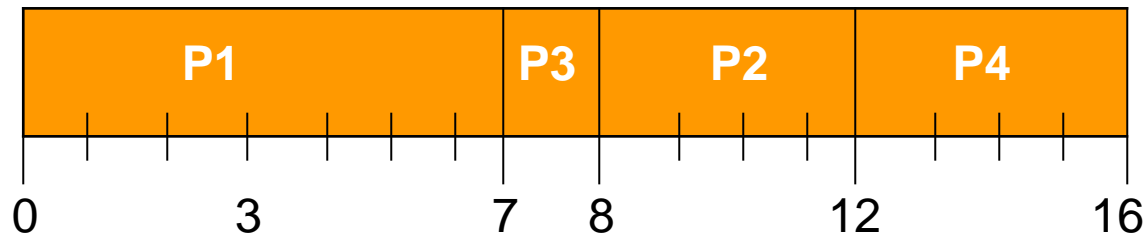
- Lựa chọn tiến trình có thời gian thực hiện ngắn nhất để phục vụ trước;
- Nguyên tắc không hoán đổi;
- Việc lập lịch được thực hiện sau khi tiến trình kết thúc.



Giản đồ Gantt khi định thời theo SJF:

VÍ DỤ

Tiến trình	Thời điểm đến	Thời gian thực thi (ms)
P1	0	7
P2	2	4
P3	4	1
P4	5	4



Thời gian đợi trung bình = $(0 + 6 + 3 + 7) / 4 = 4$.



3. CÁC THUẬT TOÁN LẬP LỊCH

3.3. GIẢI THUẬT SJF (Shortest Job First)

ƯU VÀ NHƯỢC ĐIỂM CỦA GIẢI THUẬT SJF

- Ưu điểm:
 - SJF tối ưu trong việc giảm thời gian đợi trung bình;
 - SJF nhanh chóng loại bỏ các tiến trình ngắn, giảm số tiến trình trong dòng xếp hàng sẵn sàng.
- Nhược điểm: Cần phải ước lượng thời gian cần CPU tiếp theo của tiến trình;

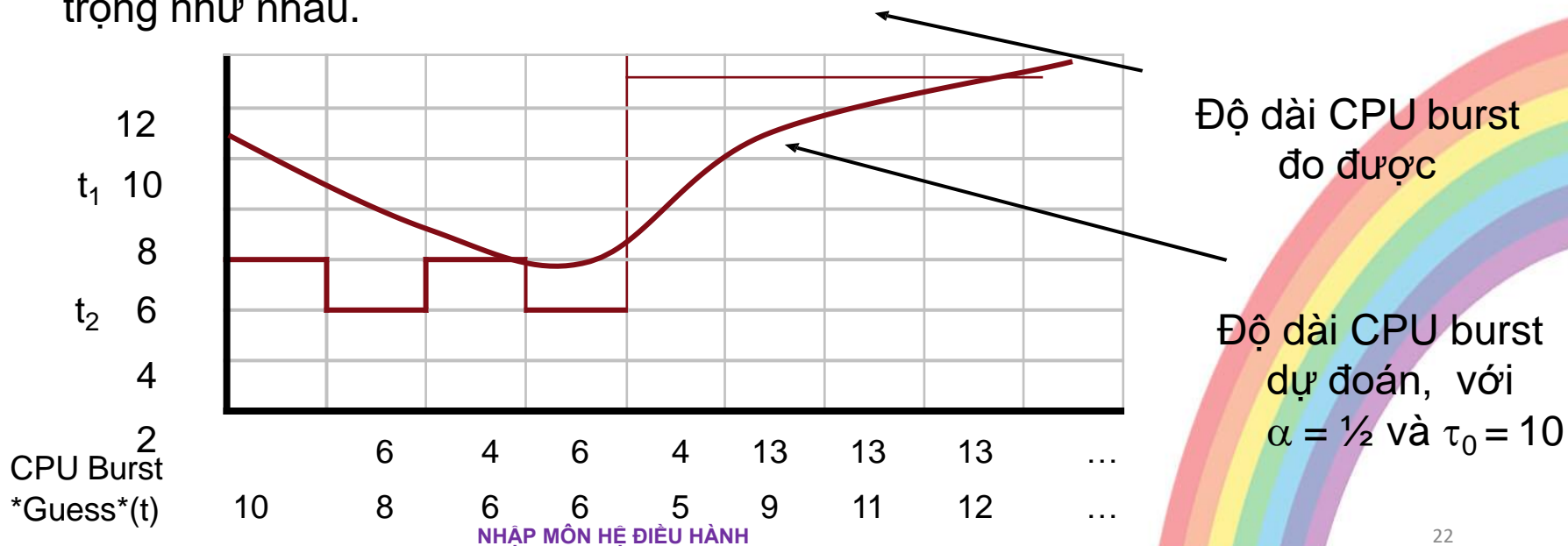


3. CÁC THUẬT TOÁN LẬP LỊCH

3.4. ƯỚC LƯỢNG THỜI GIAN CẦN CPU

Thường dùng là sử dụng trung bình hàm mũ (exponential averaging):

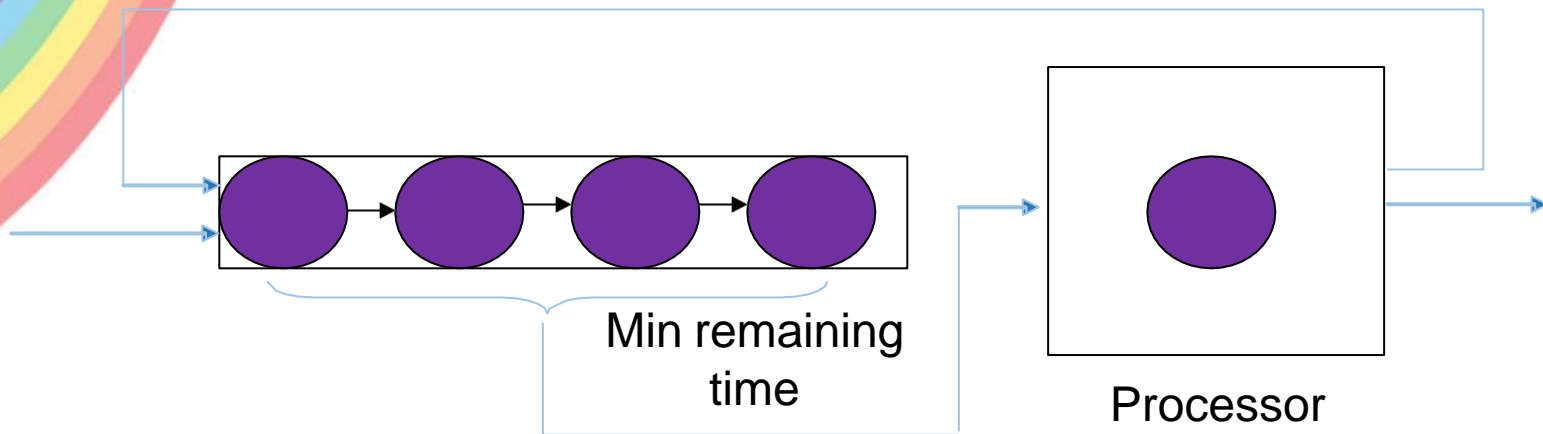
- $\tau_{n+1} = \alpha t_n + (1 - \alpha) \tau_n; 0 \leq \alpha \leq 1;$
- $\tau_{n+1} = \alpha t_n + (1 - \alpha) \alpha t_{n-1} + \dots + (1 - \alpha)^j \alpha \tau_{n-j} + \dots + (1 - \alpha)^{n+1} \alpha \tau_0;$
- Trong đó:
 - t_n : Độ dài CPU burst đo được;
 - τ_n : Thời gian ước lượng cần cho CPU tiếp theo.
- Nếu chọn $\alpha = \frac{1}{2}$ thì có nghĩa là trị đo được t_n và trị dự đoán τ_n được xem quan trọng như nhau.





3. CÁC THUẬT TOÁN LẬP LỊCH

3.5. GIẢI THUẬT SRT (Shortest Rest Time)



- Nguyên tắc có hoán đổi;
- Ưu tiên tiến trình có thời gian còn lại ít nhất;
- Việc lập lịch CPU được thực hiện ngay cả khi có quá trình mới đến hệ thống.

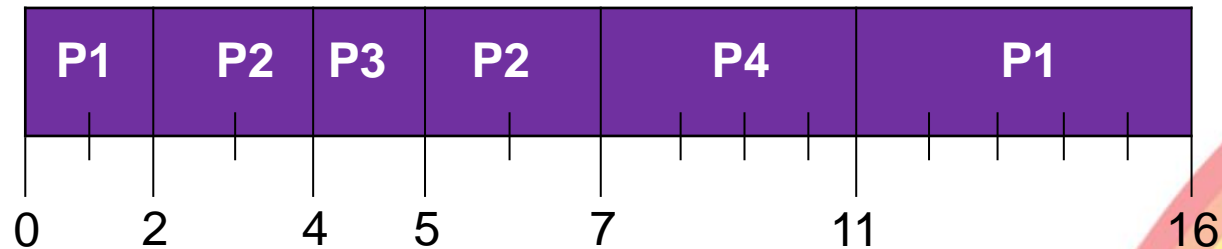


Hướng dẫn:

VÍ DỤ

Tiến trình	Thời điểm đến	Thời gian thực thi (ms)
P1	0.0	7
P2	2.0	4
P3	4.0	1
P4	5.0	4

- Giản đồ Gantt khi định thời theo SRT:



- Thời gian đợi trung bình = $(9 + 1 + 0 + 2)/4 = 3$;
- Nhận xét: Giải thuật SRT tốt hơn giải thuật không hoán đổi SJF.



3. CÁC THUẬT TOÁN LẬP LỊCH

3.5. GIẢI THUẬT SRT (Shortest Rest Time)

ƯU VÀ NHƯỢC ĐIỂM CỦA GIẢI THUẬT SRT

- Ưu điểm:
 - Tránh trường hợp các tiến trình có thời gian thực thi dài độc chiếm CPU;
 - Có thời gian quay vòng tốt hơn SJF;
 - Tiến trình có thời gian thực thi ngắn có độ ưu tiên cao.
- Nhược điểm:
 - Phải dự đoán được chính xác thời gian còn lại của tiến trình;
 - Tăng chi phí tính toán: Chi phí chuyển đổi ngữ cảnh.

3. CÁC THUẬT TOÁN LẬP LỊCH



3.5. GIẢI THUẬT SRT (Shortest Rest Time)

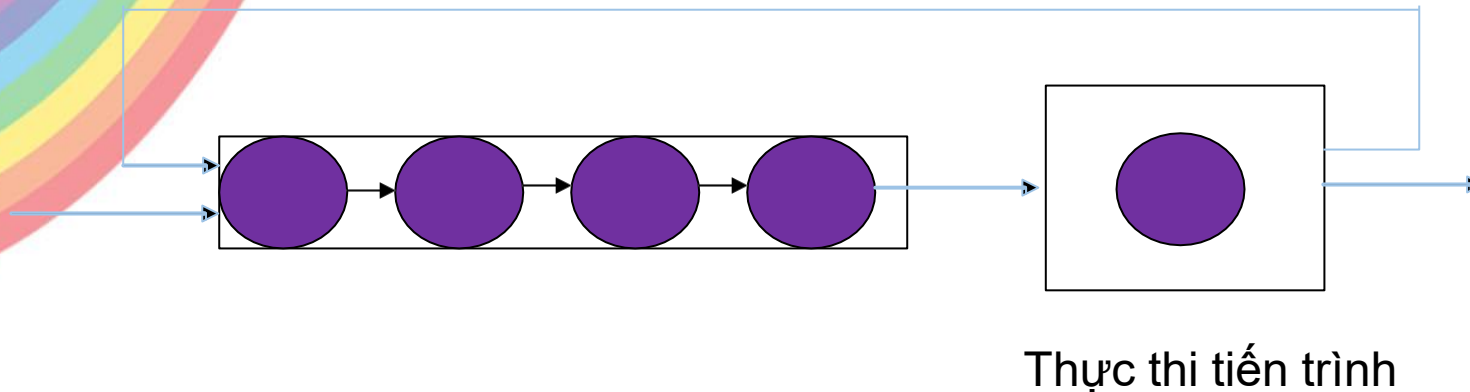
MỨC ƯU TIÊN

- Để giảm thời gian chờ đợi trung bình, hệ thống có thể xác định số thứ tự ưu tiên điều độ cao cho tiến trình có khả năng sử dụng ít thời gian hoạt động;
- Mỗi tiến trình sẽ được gán một độ ưu tiên;
- CPU sẽ được cấp cho tiến trình có độ ưu tiên cao nhất;
- Lập lịch sử dụng mức ưu tiên có thể:
 - Mức ưu tiên tĩnh: Được gán ngay khi tạo ra và không thay đổi trong suốt quá trình;
 - Mức ưu tiên động: Có thể thay đổi khác với mức ưu tiên khởi tạo ban đầu.
- Trong SJF và SRT thứ tự ưu tiên phụ thuộc vào thời gian thực hiện chương trình mà người sử dụng khai báo.
- Mức ưu tiên phụ thuộc:
 - Yêu cầu về bộ nhớ;
 - Số lượng file được mở;
 - Tỷ lệ thời gian dùng cho I/O trên thời gian sử dụng CPU;
 - Các yêu cầu bên ngoài ví dụ như: số tiền người dùng trả khi thực thi công việc

3. CÁC THUẬT TOÁN LẬP LỊCH



3.6. GIẢI THUẬT RR (Round Robin)



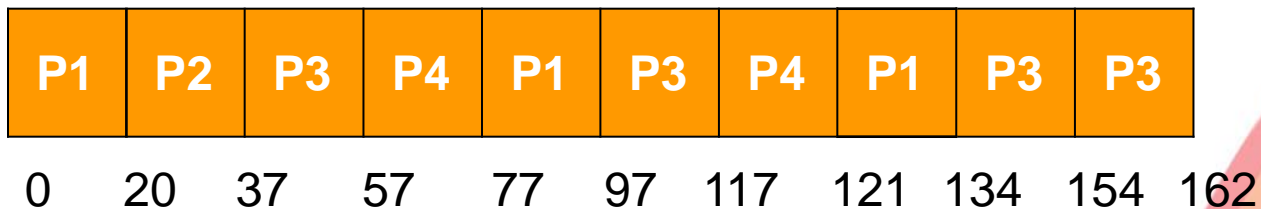
- Nguyên tắc FIFO nhưng có hoán đổi;
- Khoảng thời gian tối đa cho phép (thường 10 - 100 ms) (Quantum Time - q) được đảm bảo bằng việc sử dụng ngắt thời gian;
- Tiến trình đang chạy hết thời gian được chuyển về cuối của hàng đợi Ready.



VÍ DỤ

Tiến trình	Thời gian thực thi (ms)
P1	53
P2	17
P3	68
P4	24

- Thời gian lượng tử = 20 ms
- Giải đồ Gantt:



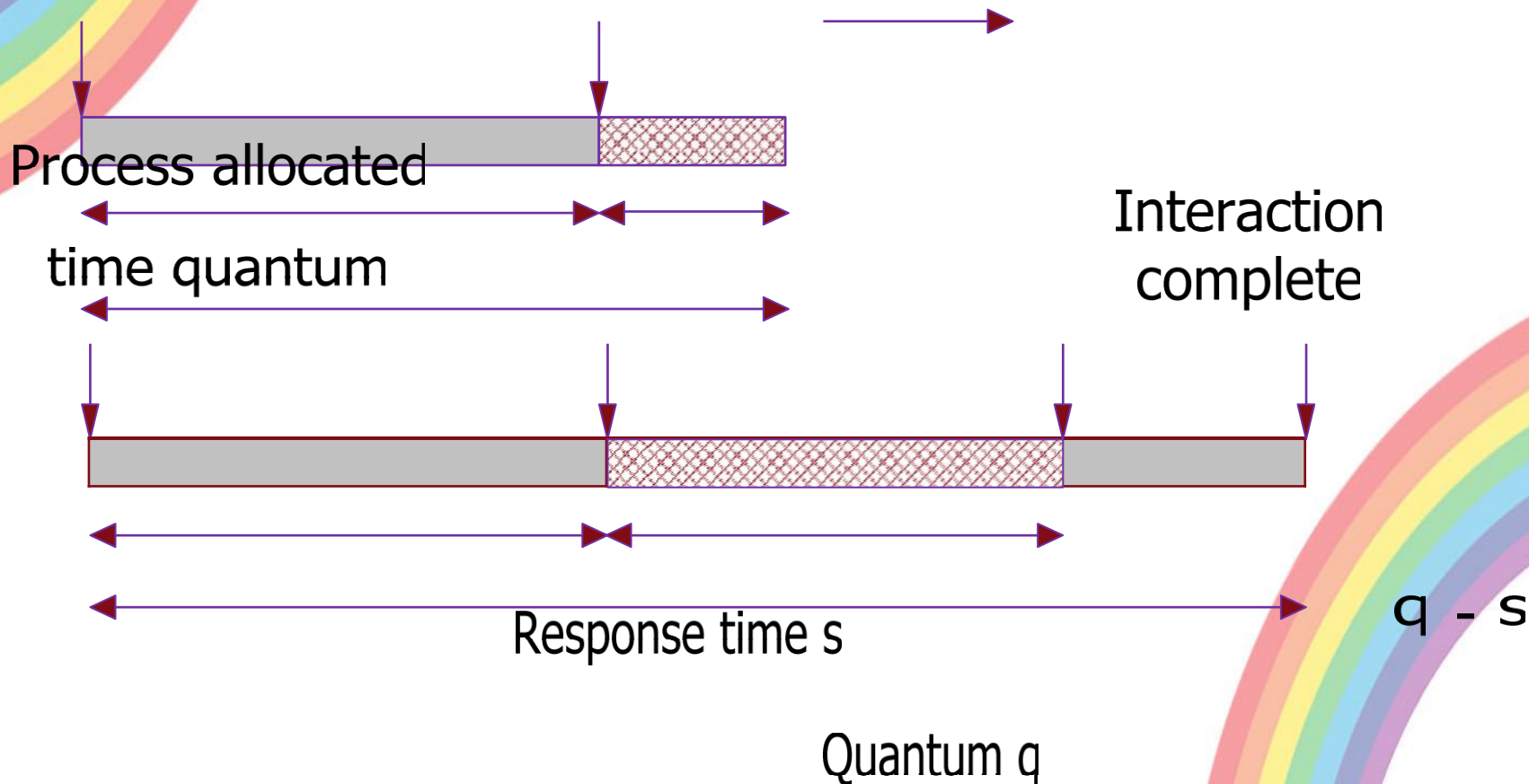
Thường có thời gian quay vòng cao hơn SJF, nhưng lại có thời gian đáp ứng tốt hơn.

3. CÁC THUẬT TOÁN LẬP LỊCH



3.6. GIẢI THUẬT RR (Round Robin)

GIÁ TRỊ CỦA LƯỢNG TỬ CHO GIẢI THUẬT RR



3. CÁC THUẬT TOÁN LẬP LỊCH

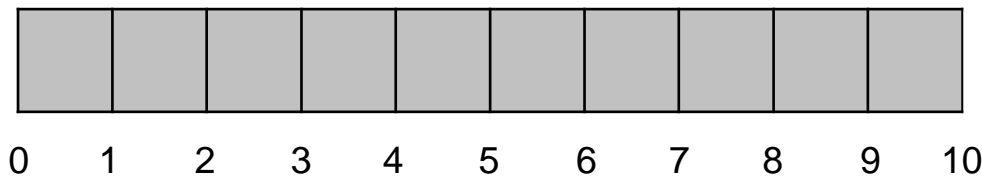
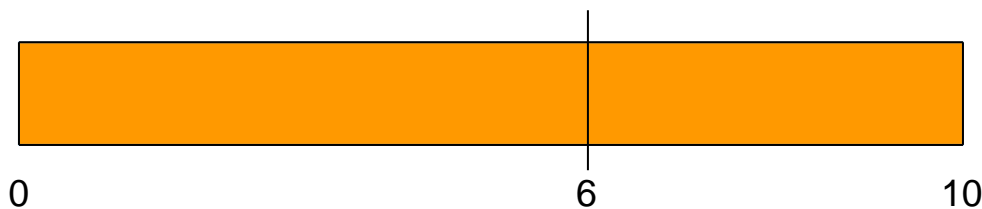
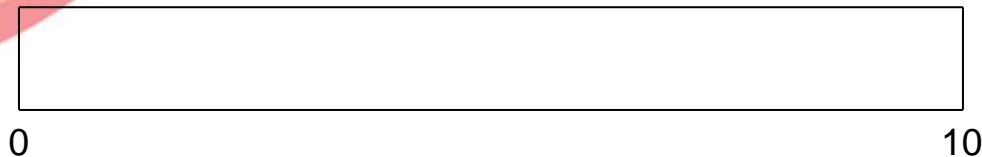


3.6. GIẢI THUẬT RR (Round Robin)

THỜI GIAN LƯỢNG TỬ VÀ CHUYỂN ĐỔI NGỮ CẢNH

Quantum time càng nhỏ thì càng có nhiều lần chuyển ngữ cảnh (Context Switch)

Process time = 10



Số lần ngưng/ tiếp tục quá trình

Quantum

Context Switches

12

0

6

1

1

9

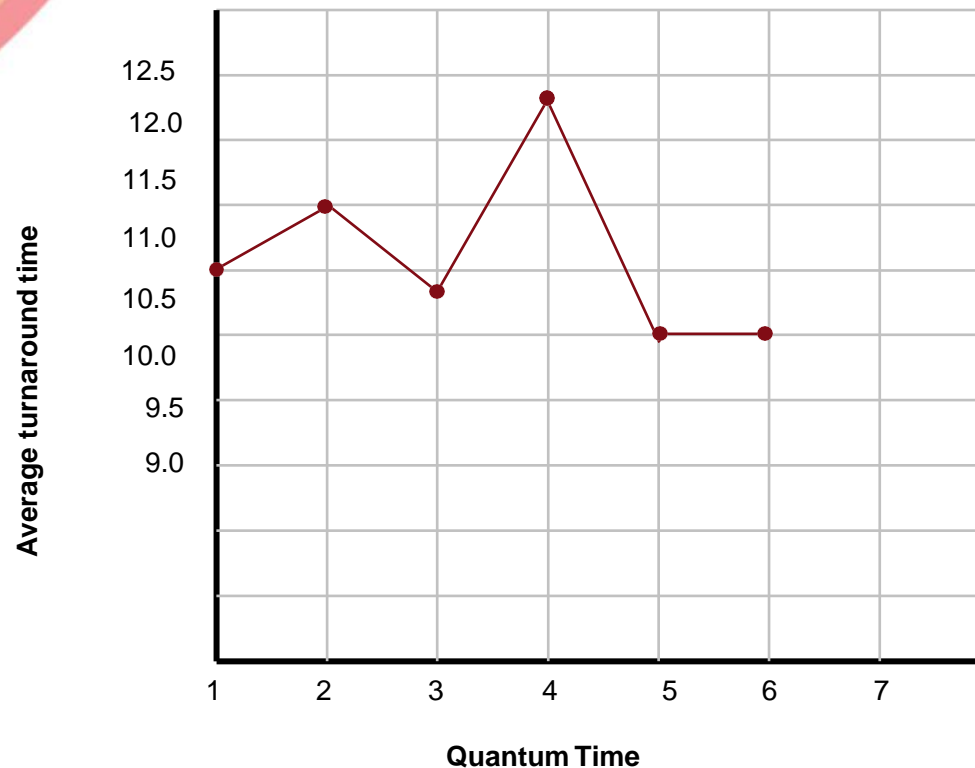
3. CÁC THUẬT TOÁN LẬP LỊCH



3.6. GIẢI THUẬT RR (Round Robin)

THỜI GIAN QUAY VÒNG VỚI THỜI GIAN LƯỢNG TỬ

Thời gian quay vòng trung bình (Average Turnaround Time) không chắc sẽ được cải thiện khi Quantum lớn.



Process	Time
P1	6
P2	3
P3	1
P4	7

3. CÁC THUẬT TOÁN LẬP LỊCH



3.6. GIẢI THUẬT RR (Round Robin)

THỜI GIAN QUAY VÒNG VỚI THỜI GIAN LƯỢNG TỬ

Quantum Time và thời gian cho chuyển đổi ngữ cảnh:

- Nếu Quantum Time quá nhỏ thì chi phí chuyển đổi ngữ cảnh chiếm phần lớn thời gian và năng lực tính toán của hệ thống;

Với Quantum Time = 10 ms và thời gian cho chuyển đổi ngữ cảnh = 5ms, như vậy phí tổn chiếm $5 / 10 = 50\%$.

- Nếu Quantum Time lớn trở thành giải thuật FIFO:
 - Với Quantum = 500 ms, thì phí tổn chỉ còn 1%.
 - Nhưng nếu có nhiều người sử dụng trên hệ thống và thuộc loại Interactive thì sẽ thấy đáp ứng rất chậm;
 - Tùy thuộc vào tập công việc mà lựa chọn Quantum Time;
 - Khoảng thời gian nên lớn trong tương quan so sánh với thời gian cho chuyển ngữ cảnh.

3. CÁC THUẬT TOÁN LẬP LỊCH



3.7. HRRN (Highest Response Ration Next)

- Cải tiến giải thuật SJF;
- Định thời theo kiểu **không hoán đổi**;
- Độ ưu tiên của quá trình được tính theo công thức:

$$p = (t_w + t_s)/t_s$$

Trong đó:

- t_w : Thời gian chờ;
- t_s : Thời gian thực thi.
- Quá trình có độ ưu tiên lớn nhất được phục vụ;
- Độ ưu tiên động, tính lại khi có quá trình kết thúc.



VÍ DỤ

- Khi P1 kết thúc, hệ thống lập lịch lại;
- Độ ưu tiên:

➤ $P2: (6 + 4) / 4 = 2.5$

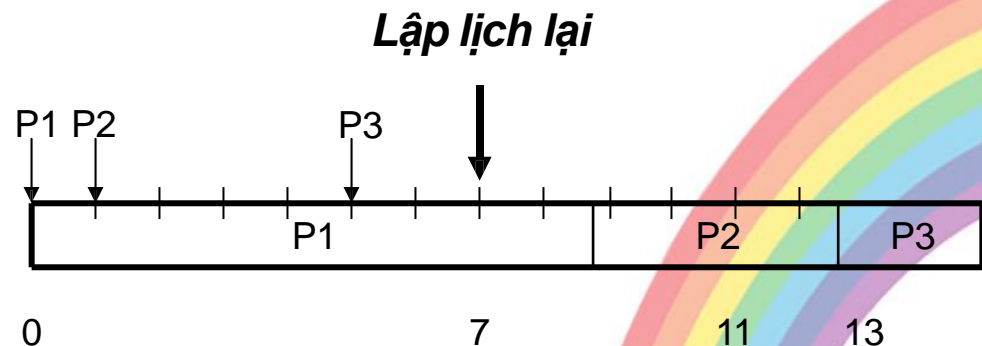
➤ $P3: (2 + 2) / 2 = 2$

→ P2 được ưu tiên.

- Thứ tự lập lịch:

$P1 \rightarrow P2 \rightarrow P3$

Quá trình	Thời gian đến	Thời gian thực thi (CPU burst time - giây)
P1	0	7
P2	1	4
P3	5	2



3. CÁC THUẬT TOÁN LẬP LỊCH



3.8. HÀNG ĐỢI NHIỀU MỨC (Multilevel Feedback Queue)

- Trường hợp các quá trình có thể được phân thành nhóm.
 - Ví dụ: Interactive và Batch.
- Hàng đợi Ready sẽ được chia thành nhiều hàng đợi riêng rẽ.
 - Ví dụ: Mức trên (cho công việc cần giao tiếp - Interactive), mức dưới (cho công việc dạng bó - Batch).
- Mỗi hàng đợi sẽ có giải thuật lập lịch riêng.
 - Ví dụ: Mức trên dùng RR, mức dưới dùng FCFS.
- Lập lịch cần phải thực hiện giữa các hàng đợi với nhau:
 - Theo cách cố định (Fixed Priority Scheduling):
 - Ví dụ: Phục vụ tất cả các tiến trình của mức trên rồi mới đến mức thấp;
 - Có khả năng xảy ra trì hoãn vô hạn định (Starvation).
 - Chia thời gian (Time Slice) – mỗi hàng đợi sẽ được lấy một khoảng sử dụng CPU nhất định để lập lịch cho các tiến trình của mình.
- Ví dụ: 80% cho mức trên (dùng RR); 20% cho mức thấp (dùng FCFS).



VÍ DỤ PHÂN NHÓM CÁC QUÁ TRÌNH

Độ ưu tiên cao nhất



Độ ưu tiên thấp nhất

3. CÁC THUẬT TOÁN LẬP LỊCH



3.8. HÀNG ĐỢI NHIỀU MỨC (Multilevel Feedback Queue)

MỨC ƯU TIÊN TRONG HÀNG ĐỢI NHIỀU MỨC

Ưu tiên tiến trình thường xuyên thực hiện thao tác I/O:

- Lượng tử chọn: Đủ để tiến trình thực hiện xong tính toán và yêu cầu thao tác I/O;
- Tiến trình vẫn được đưa vào cuối hàng đợi, gán với mức ưu tiên cao => nhanh chóng được phục vụ;
- Tiến trình thực hiện tính toán nhiều: Đưa tiến trình vào hàng đợi mức trên => được phục vụ
- Kết thúc lượng tử: Được đưa vào hàng đợi mức dưới với mức ưu tiên thấp.
- Được phục vụ khi: Ở đầu hàng đợi và không còn tiến trình nào trong các hàng đợi mức trên.
- Có thể bị ngắt nếu có tiến trình mới xuất hiện thuộc hàng đợi mức trên.

3. CÁC THUẬT TOÁN LẬP LỊCH



3.8. HÀNG ĐỢI NHIỀU MỨC (Multilevel Feedback Queue)

MỨC ƯU TIÊN TRONG HÀNG ĐỢI NHIỀU MỨC

- Tiến trình chuyển từ pha tính toán nhiều sang pha thực hiện I/O:
 - Hệ thống ghi lại thời gian các lần tiến trình chiếm bộ xử lý;
 - Khi chuyển pha hệ thống có thể phát hiện nhanh;
 - Đưa tiến trình đó vào hàng đợi thích hợp.
- Ưu điểm:
 - Phân tách các tiến trình theo thời gian sử dụng bộ xử lý;
 - Hệ thống linh động, tự điều chỉnh hoạt động và lợi ích đạt được cân bằng với chi phí.
- Nhược điểm: Chi phí lớn.





TÓM LƯỢC CUỐI BÀI

- Sự thực thi của một tiến trình;
- Bộ định thời chọn một tiến trình từ hàng đợi sẵn sàng;
- Dispatcher thực hiện chuyển ngữ cảnh;
- Các tiêu chí định thời (thường xung đột nhau);
- Độ lợi CPU, thời gian chờ, thời gian đáp ứng, thông năng,...
- Các giải thuật định thời;
- FCFS, SJF, SRT, HRRN, RR, Hàng đợi nhiều mức....