CSC10007 – HỆ ĐIỀU HÀNH BÀI TẬP LÝ THUYẾT

BTLT-02 – ĐIỀU PHỐI CPU

Đáp án bài 5

0																										х
r2																		х		х				х		
r1				х															x		х					
f4					х	х	х																			
f3											х	х	х	х												
f2		x	х					х		х					х	х						х	х			
f1	х								х								х								х	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Queue	f1 f2 o	f2 o	f2 o	r1 o	r1 r2 f4 o	r1 r2 f4 f3 o	r1 r2 f4 f3 o	f2 r1 r2 f3 o	f2 r1 r2 f3 f1 o	r1 r2 f3 f2 o	r1 r2 f3 o	r1 r2 f3 o	r1 r2 f3 o	r1 r2 f3 o	f2 r1 r2 o	f2 r1 r2 o	f1 r1 r2 o	r1 r2 o	r1 r2 o	r1 r2 o	r1 r2 o	f2 r2 o	f2 r2 o	r2 o	f1 0	0

Đáp án bài 6

→PRIORITY INVERSION

D																		R										R					
С																																	
В			R	R								R		R															R	R			
Α																																	
	0	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	18	1	2	2	2	2	2	2	2	2	2	2	30	3	3
											0	1	2	3	4	5	6	7		9	0	1	2	3	4	5	6	7	8	9		1	2
	AB	В				D	Α				В		Α			D	D	D	Α	С	В	₿			Α	С		D			Α		
	D	D					D												С	D	С	С			С	D					В		
																			D		D	D			D								

CSC10007 – HỆ ĐIỀU HÀNH BÀI TẬP LÝ THUYẾT

BTLT-03: ÔN TẬP THI GIỮA KỲ

Đáp án Bài 7: Cho hệ thống điều phối với thuật toán SJF không độc quyền có 3 tiến trình với thời điểm đến (Arrival) và các chu kỳ CPU (Burst) và I/O (Burst) cho hai tài nguyên R1 và R2 như bảng bên dưới. R1 và R2 điều phối theo chiến lược FCFS độc quyền. Vẽ sơ đồ Gantt minh hoạ quá trình điều phối CPU và tài nguyên R1, R2 cho 3 tiến trình.

Drassa	Arrival	CPU	1/4	0	CPU	I/O			
Process	Arrivai	Burst	Resource	Burst	Burst	Resource	Burst		
P1	0	5	R1	2	2	R2	2		
P2	2	1	R1	10	1	R1	4		
P3	10	8	R2	1	-	-	-		

Sơ đồ Gantt:

R2									P1 → exit		P3→exit	
R1			P2	P2	P2	P1	P1	P2	P2 → exit			
CPU	P1	P2	P1	idle	Р3	P2	Р3	P1	P3	Р3		
	0	2	3	6	10	13	14	15	17	19	21	22

Bổ sung:

- Tính toán thời gian lưu trú (Turnarround Time:TT) cho từng tiến trình, từ đó suy ra thời gian lưu trú trung bình (TTavg).
- Tính toán thời gian chờ [CPU trong ready list] (Waiting Time-WT) cho từng tiến trình, từ đó suy ra thời gian chờ trung bình (WTavg)
 - o Trong ngữ cảnh CPU Scheduling thường hiểu là thời gian chờ CPU == Thời gian trong ready list
 - o Bài toán tổng quát:
 - Thời gian chờ trong hệ thống == Thời gian chờ CPU + Thời gian chờ nhập xuất
 - Thời gian chờ nhập xuất == Thời gian trong waiting list tương ứng của I/O resouces/devices
- Cho biết số lần hệ thống phải thực hiện chuyển đổi ngữ cảnh (context switch) cho quá trình điều phối trên (Không tính lần chuyển đổi ngữ cảnh đầu tiên khi P1 được giao CPU và lần chuyển ngữ cảnh cuối cùng khi P3 hoàn tất và hệ thống cần chọn tiến trình X khác).

Process	Arrival	Completion	TT	WT	TTavg	WTavg	Context Switch
P1	0	22	22	1 + 0 = 1			
P2	2	19	17	0 + 0	1		7
P3	10	19		3	1		

Đáp án Bài 9: Hệ thống X sử dụng thuật toán điều phối CPU với hàng đợi phản hồi đa cấp (multilevel level feedback). Hệ thống sử dụng 3 hàng đợi: Q1, Q2, và Q3 với các thuật toán điều phối khác nhau.

- Q1 dùng RR với q = 4; Q2 dùng RR với q = 8; Q3 dùng FCFS.
- Khi một tiến trình ở Q1 sử dụng hết q nhưng chưa hoàn tất sẽ chuyển xuống Q2. Tương tự, khi một tiến trình ở Q2 sử dụng hết q nhưng chưa hoàn tất sẽ chuyển xuống Q3.
- Độ ưu tiên của các hàng đợi được gán như sau: Q1 > Q2 > Q3 (Q2 được xem xét khi Q1 rỗng; tương tự, Q3 được xem xét khi Q1 & Q2 rỗng).
 Giả sử có 5 tiến trình với thời điểm vào Ready List (Arrival Time) và thời gian xử lý (CPU-Burst được mô tả như bảng bên dưới:

Process	Arrival Time	CPU Burst	Completion	TT	WT	Statistics
P1	0	10	25	25	15	Number of Context Switches: 7
P2	2	5				TTtavg:?
P3	4	7				WTavg:?
P4	6	3				
P5	8	4				

- a) Vẽ sơ đồ Gantt cho quá trình điều phối các tiến trình trên với hàng đợi phản hồi đa cấp.
- b) Tính thời gian lưu trú của từng tiến trình (TT), và thời gian lưu trú trung bình.
- c) Tính thời gian chờ của từng tiến trình (WT) và thời gian chờ trung bình.

CPU	P1	P1	P2	P2	P3	РЗ	P4	P4	P5	P5	P5	P1	P2	P3			
	0	2	4	6	8	10	12	14	15	16	17	19	25	26	29	27	30
Q1	P1	P1	P2	P2	P3	-nt-	P4	-nt-	P5								
		P2	P3	P3	P4		P5										
				P4	P5												
Q2			P1		P1	-nt-	P1	-nt-	-nt-			P1	P2	P3			
					P2		P2					P2	P3				
							P3					P3					
Q3																	

Thảo luận:

- 1) Process P1 ở hàng đợi Q2 chưa hết time slice nhưng bị thu hồi CPU bởi hàng đợi cao hơn? P1 di chuyển đi đâu?
 - P1 xem như đã hoàn tất 1 đơn vị xử lý, thời gian dư bị huỷ
 - P1 đi xuống hàng đợi có ưu tiên thấp hơn
 - P1 đi xuống cuối hàng đợi hiện tại đang ở
 - P1 được giữ lại remaining time-slice (thời gian chưa sử dụng hết của P1)
 - → cần cấu trúc dữ liệu và thuật toán quản lý thông tin về remaining time của các tiến trình (cf. SRTN)

Xây dựng thuật toán điều phối với hàng đợi đa cấp:

- Bao nhiêu hàng đợi?
- Độ ưu tiên của từng hàng đợi?
- Thuật toán điều phối cho một hàng đợi
- Tiêu chí đưa tiến trình new vào hàng đợi:
 - Cố đinh?
 - Có thể thay đổi? → Tiêu chí thay đổi: khi nào đưa xuống lower priority? Khi nào đưa lên higher priority?
- 2) P1, P2, và P3 lần lượt vào ready list tại một thời điểm t từ 3 trạng thái:
 - running (do hết time-slice),
 - blocked/waiting (do v\u00fca xong I/O)
 - new (do vừa được khởi tạo và chuyển từ job list vào thông qua long-term scheduling)

Chọn "anh" nào?

3) P1 nằm ở hàng đợi điều phối RR với q = 4,

Thực thi được 2 thì bị lấy lại CPU bởi một tiến trình khác có độ ưu tiên cao hơn? P1 đi đâu?