

CHƯƠNG 5

ĐIỀU KHIỂN TƯƠNG TRANH (tt)

NỘI DUNG

1. Điều Khiển Tranh Bằng Cơ Chế Nhãn Thời Gian (Timestamp)
2. Thuật toán sắp xếp toàn phần
3. Thuật toán sắp xếp từng phần
4. Thuật toán sắp xếp từng phần đa phiên bản

- **Đường Link you tube**

- <https://www.youtube.com/watch?v=jj4IjCH1I4U&list=PL67CJL04EcjN5PQrippgbVWz9L06a3Atz&index=2>

1- Điều Khiển Tranh Tranh Bằng Cơ Chế Nhãn Thời Gian (Timestamp)

Khái niệm nhãn thời gian

Là một con số được phát sinh bởi bộ lập lịch, gán cho các giao dịch để chỉ thời điểm bắt đầu thực hiện của các giao dịch. Nhãn thời gian có tính chất duy nhất và tăng dần.

▪ Đơn vị dữ liệu (Granule)

Đơn vị dữ liệu có thể là trang (page), bảng (table), dòng dữ liệu

▪ Nhãn thời gian của đơn vị dữ liệu

Nhãn thời gian của đơn vị dữ liệu chính là nhãn thời gian của giao dịch cuối cùng có truy cập đến đơn vị dữ liệu đó hoặc là nhãn thời gian cao nhất trong số các giao dịch có truy cập đến đơn vị dữ liệu đó.

T1 (100)	T2 (200)	T3 (250)	tA
Read A			$tA=tT1=100$
	Read A		$tA=tT2=200$
		Read A	$tA=tT3=250$

2 - Thuật toán sắp xếp toàn phần

Thuật toán này không phân biệt thao tác đọc hay ghi.

i, j là nhãn thời gian của giao dịch **T_i, T_j**.

tg là nhãn thời gian của **đơn vị dữ liệu g**.

tg = 0, vì ban đầu chưa có giao dịch nào truy cập.

* Thuật toán đọc:

Procedure **Read** (**T_i**, **g**)

Begin

 If **tg ≤ i** then

 Thực hiện thao tác đọc

tg := i

 Else

 Abort (**T_i**)

End Procedure

* Thuật toán viết :

Procedure **Write** (**T_i**, **g**)

Begin

 If **tg ≤ i** then

 Thực hiện thao tác viết

tg := i

 Else

 Abort (**T_i**)

End Procedure

Ví dụ 1

T1	T2	T3	tA	tB	tC
tT1=200	tT2=150	tT3=175	0	0	0
	Read A				
		Read C			
Read B					
Write B					
Write A					
		Write A			

Kết luận:
T1 thành công.
T2, T3 bị Rollback và bắt đầu lại với ntg mới

Dùng thuật toán sắp xếp toàn phần để điều khiển tương tranh cho lịch thao tác bên?

Kết quả

T1	T2	T3	tA	tB	tC
tT1=200	tT2=150	tT3=175	0	0	0
	Read A		150	0	0
		Read C	150	0	175
Read B			150	200	175
Write B			150	200	175
Write A			200	200	175
	Write C		200	200	T2 rollback
		Write A	T3 rollback	200	175

2- Thuật toán sắp xếp toàn phần

Ví dụ 2:

T1 (100)	T2 (120)	tA
Read A		=100
	Read A	=120
	Read A	=120
Read A		nên T1 phải rollback và bắt đầu lại với timestamp mới

Nhận xét:

- Hai giao dịch T1, T2 truy cập đồng thời trên CSDL, nhưng chỉ thao tác đọc
→ không tranh chấp mà T1 phải bị rollback.
- Đây là **nhược điểm** của thuật toán sắp xếp toàn phần.
→ **Để tránh nhược điểm này, ta tìm hiểu qua thuật toán sắp xếp từng phần**

Mỗi đơn vị dữ liệu có 2 nhãn thời gian tr , tw

- ✓ **tr** là nhãn thời gian của giao dịch già nhất có nhãn thời gian lớn nhất trong số các giao dịch thực hiện **thao tác đọc** trên đơn vị dữ liệu này.
- ✓ **tw** là nhãn thời gian của giao dịch già nhất có nhãn thời gian lớn nhất trong số các giao dịch thực hiện **thao tác viết** trên đơn vị dữ liệu này.

Ở thời điểm ban đầu $tr = 0$, $tw = 0$.

** Thuật toán đọc:*

Procedure **Read**(**Ti** , g)

Begin

If **tTt** \geq **twg** then

Thực hiện thao tác đọc

trg = max (trg , tTi)

Else

Rollback Ti và bắt đầu lại với
timesrtamp mới

End Procedure

** Thuật toán viết :*

Procedure **Write**(**Ti** , g)

Begin

If (**t_{rg}** \leq tTi) and (twg \leq tTi) then

Thực hiện thao tác viết

Twg = tTi

Else

Rollback Ti và bắt đầu lại với
timesrtamp mới

End Procedure

Ví dụ 1: Dùng thuật toán sắp xếp từng phần để điều khiển tương tranh cho lịch thao tác sau?

T1(100)	T2(120)	t_{rA}	t_{wA}
Read A		100	0
	Read A	120	0
	A = A+1	120	0
A = A+1		120	0
	Write A	120	120
Write A		120	120, T1 rollback

Kết luận: T2 thành công

T1 bị Rollback và bắt đầu lại với ntg mới

Ví dụ 2:

T1 (100)	T2 (200)	Nhận xét
(1) Read A		Thuật toán sắp xếp toàn phần : T1 bị rollback ở (3) Thuật toán sắp xếp từng phần : không có giao dịch nào bị rollback
	(2) Read A	
(3) Read A		

T1 (100)	T2 (200)	Nhận xét
(1) Read A		Thuật toán sắp xếp từng phần : T1 bị rollback ở (3) Thuật toán sắp xếp toàn phần : T1 bị rollback ở (3)
	(2) Write A	
(3) Read A		

❖ **Nhận xét:** trong thuật toán sắp xếp từng phần, số lượng giao dịch bị Rollback ít hơn trong thuật toán sắp xếp toàn phần.

Nếu có 2 giao dịch cùng thực hiện thao tác “Đọc” thì không gây ra ñụng ñộ.

3- Thuật toán sắp xếp từng phần đa phiên bản

- Mỗi đơn vị dữ liệu được truy cập có 1 tập tr, tw và giá trị
 - Khi thực hiện 1 thao tác cập nhật trên 1 đơn vị dữ liệu thì phải tạo ra 1 phiên bản mới cho đơn vị dữ liệu đó (giúp các giao dịch có tính cô lập)
 - Ở thời điểm ban đầu $tr = tw = 0$
- > 2 thuật toán đọc và viết trên đơn vị dữ liệu

*** Thuật toán đọc:**

Procedure **Read**(**Ti** , g)

Begin

J:= số phiên bản cuối cùng của g

While twg **J**> t**Ti** do **J**:=**J**-1

Thực hiện thao tác đọc g
trên phiên bản thứ **J**

trgi = max (trg i, tTi)

End Procedure

*** Thuật toán viết :**

Procedure **Write**(**Ti** , g)

Begin

J:= số phiên bản cuối cùng của g

While twg **J**> t**Ti** do **J**:=**J**-1

If (trgi > t**Ti**) then

Rollback **Ti** và bắt đầu lại

Else

Thực hiện thao tác viết

Tạo thêm 1 phiên bản mới **j +1**
cho g (**TwgJ +1 := tTi**)

End Procedure

Ví dụ 1:

T1 (100)	T2 (110)
Read A	
	Write A
Read A	

Nếu 1 phiên bản có cấu hình như sau

Tr		Tw
Phiên bản thứ J		

Các phiên bản cho ví dụ trên với đơn vị dữ liệu là A

100		0
Phiên bản thứ 1		

100		110
Phiên bản thứ 2		

Ở thời điểm 3, thao tác T1 read A. Vì $t_{T1} = 100 < t_{wA} = 110$ nên thao tác T1 đọc lên phiên bản trước đó là phiên bản 1

Ví dụ 2:

T1 (100)	T2 (200)
Read A	
	Read A
	Write B
Read B	
Write A	

Các phiên bản cho ví dụ trên với **đơn vị dữ liệu là A:**

100		0	200		0	200>100	
Phiên bản thứ 1			Phiên bản thứ 1			Roll back T1	

Các phiên bản cho ví dụ trên với **đơn vị dữ liệu là B:**

0		200	100		200
Phiên bản thứ 1			Phiên bản thứ 1		

Giao tác T chỉ đọc giá trị của phiên bản do nó hay những giao tác trước nó cập nhật. T không bao giờ đọc giá trị do các phiên bản sau nó cập nhật.

Thao tác đọc không bao giờ Rollback, thao tác ghi thì có thể bị Rollback hoặc chèn thêm 1 phiên bản mới.

- **Câu hỏi ôn tập**

- Có bao nhiêu cách để điều khiển tương tranh? Đó là những cách nào?
- So sánh 2 thuật toán sắp xếp: toàn phần, từng phần
- So sánh 2 thuật toán sắp xếp: từng phần, từng phần đa phiên bản

- **Bài tập nhóm**

- Các sinh viên chia nhóm. Mỗi nhóm 3 sinh viên thực hiện yêu cầu sau:
- Sinh viên 1: Lập trình giải quyết bài toán Sắp xếp toàn phần
- Sinh viên 2: Lập trình giải quyết bài toán Sắp xếp từng phần
- Sinh viên 3: Lập trình cài đặt bài toán Sắp xếp từng phần đa phiên bản

- **Tóm tắt**

Có 3 thuật toán điều khiển tương tranh bằng cơ chế nhả thời gian:

- Thuật toán sắp xếp toàn phần
- Thuật toán sắp xếp từng phần
- Thuật toán sắp xếp từng phần đa phiên bản