14.5: از آنجایی که هر اسکجول conflict-serializable، view serializable هم می باشد، چرا ما به جای view serializable تاکید بر conflict-serializableمی کنیم؟

A: بسیاری از پروتکل های کنترل همروندی ( پروتکل هایی برای تضمین اینکه اسکجول ها تنها serializable تولید می شوند) در عمل برمبنای conflict-serializable استفاده می شوند—آنها عملا تنها به یک زیر مجموعه از اسکجول های conflict serializable اجازه می دهند.تست کردن شکل کلی view serializability بسیلر گران است، و تنها یک شکل بسیار محدود از آن در کنترل همروندی استفاده می شود.

14.6: نمودار تقدم 14.16 را در نظر بگیرید.آیا اسکجول مربوطه conflict serializable است؟پاسخ خود را توضیح دهید.



شکل 14.16: گراف اولویت برای انجام تمرین 14. 6

A: اسکجول مربوطه serializable می باشد زیرا این گراف بدون دور است.یک اسکجول ممکن است به وسیله توپولوژی مرتب سازی به دست آمده باشد، ای هست،T5,T4,T3,T2,T1

14. 7: اسکجول cascadeless چیست؟چرا cascadeless بودن مطلوب است؟آیا شرایط مطلوبی وجود دارد که تحت آن اجازه بدهیم اسکجول بدون cascadeless باشد؟

A: یک اسکجول cascadeless که در آن، برای هر جفت از تراکنش های Ti و Tj، که Tj دیتاهایی که توسط Ti قبلا نوشته شده خوانده است، به نظر می رسد پایان عمل Ti قبل از عمل خواندن Tj می باشد.اسکجول های Cascadeless مطلوب هستند زیرا خراب شدن یک تراکنش منجر به abort شدن بقیه برنامه ها نمی شود. البته این به قیمت کم کردن هم روندی می باشد.اگر شکست به ندرت اتفاق می افتد، به طوری که ما می توانیم هزینه cascading abort را برای افزایش همروندی بدهیم اسکجول بدون cascadeless می تواند مطلوب باشد.

14. 8: زمانی یک ناهنجاری بروزرسانی از دست رفته (Lost Update) اتفاق می افتد که گفته می شود اگر یک تراکنش Tj یک دیتا آیتمی را بخواند، سپس یک تراکنش دیگر Tk آن دیتا آیتم را بنویسد( احتمالا اساس خواندن قبلی است) و بعد از آن Tj همان داده را می نویسد.بروزرسانی انجام شده توسط Tk از بین رفته می باشد،پس از پایان به روز رسانی توسط Tj مقادیر نوشته شده توسط Tk نادیده گرفته می شود.

a. یک مثال برای نمایش دادن ناهنجاری Lost Update بیاورید؟

b. یک اسکجول برای نشان دادن اینکه ممکن است به وسیله **read committed** در سطح isolation ناهنجاری Lost Update رخ دهد مثال بزنید؟

c. توضیح دهید چرا ناهنجاری Lost Update به وسیله تکرار خواندن در سطح isolation ممکن نیست؟

A:

a. یک مثال برای نشان دادن ناهنجاری Lost Update



در اسکجول بالا، مقدار نوشته شده توسط تراکنش T2 از بین رفته است به دلیل نوشتن تراکنش T1

b. ناهنجاری Lost Update در سطح Read Committed Isolation



قفل گذاری در اسکجول بالا committed read در سطح isolation را تضمین کرده است.مقدار نوشته شده توسط تراکنش T2 به علت نوشتن T1 از بین رفته است.

c. در تکرار خواندن سطح isolation ناهنجاری Lost Update ممکن نیست.در تکرار خواندن سطح isolation یک تراکنش T1 یک دیتا آیتم X ای را می خواند، یک Lock S روی X نگه می دارد تا اینکه کارش تمام شود.این کار باعث می شود تراکنش جدیدی مانند T2 نتواند مقدار جدیدی از X را بنویسد( که نیاز به Lock-X دارد)تا زمانی که T1 به پایان برسد.این کار T1 و T2 را مجبور به اجرای serialization می کند بنابراین مقدار نوشته شده توسط T2 از بین نمی رود.