

14.1 (فرض کنید یک پایگاه داده وجود دارد که هیچ وقت دچار خطا (fail) نمی شود ، آیا در این حالت نیاز به recovery manager داریم؟؟

پاسخ: حتی در این حالت recovery manager مورد نیاز است برای عملیاتی مثل roll back کردن یا Abort کردن تراکنش

14.2) فرض کنید شما یک سیستم فایل دارید:

الف) چه مراحل طی ایجاد یا حذف فایل و زمان نوشتن داده در یک فایل با آن روبرو می شوید ؟
پاسخ : چندین مراحل در ایجاد فایل وجود دارد. یک ناحیه برای فایل مورد نظر در نظر گرفته می شود ، یک عدد یکتا i برای فایل در نظر گرفته می شود، و مقدار نود i درون لیست i قرار می گیرد .
مراحل حذف نیز برعکس درج است.

ب) اصطلاحات atomicity و durability در طی مراحل حذف و ایجاد فایل و نوشتن داده در فایل را توضیح دهید؟

پاسخ: برای سیستم فایل در یونیکس پایداری به دلایل واضحی مورد نیاز است اما Atomicity به دلیلی اینکه فایل سیستم تراکنش رو پشتیبانی نمی کند مورد نیاز نیست . با این حال در زمان پیاده سازی سیستم فایل بیشتر عملیات داخلی در فایل سیستم باید مفاهیم تراکنشی داشته باشند . تمام مراحل در درج و حذف فایل باید Atomic باشند در غیر اینصورت فایل های بدون منبع یا فضاهای خالی بدون استفاده در فایل سیستم داریم

14.3 (در پیاده سازی سیستم پایگاه داده باید بیشتر به بحث خاصیت اسیدی (ACID) توجه داشته باشیم تا به پیاده سازی سیستم فایل . چرا باید اینگونه باشد؟

پاسخ: سیستم پایگاه داده معمولا وظایف های بحرانی انجام می دهد که نیاز به Atomic بودن و پایداری دارد مثلا تراکنش های بولی ، سیستم رزرو صندلی و .. ، از این رو خاصیت اسیدی باید تضمین کند ولی در مقابل کسانی از فایل سیستم استفاده می کنند قصد پرداخت هزینه بابت پشتیبانی از خاصیت اسیدی را ندارند.

14.4) جمله زیر را توجیه کنید ، اجرای همروند تراکنش ها مهتر است زمانی که داده باید از دیسک (سرعت کم) واکنشی شود یا زمانی که تراکنش ها طولانی اند و اجرای همروند زمانی اهمیت کمتری دارد که داده از حافظه رم خوانده می شود و یا اندازه تراکنش خیلی کوچک است .

پاسخ :

زمانی که اندازه تراکنش طولانی باشد یا زمانی که نیاز به خواندن از دیسک است زمان زیادی برای تمام شدن تراکنش مورد نیاز است ، در نبود همروندی مجبور به انتظار زیادی هستیم میانگین زمان پاسخ نیز افزایش می یابد . هم چنین زمانی که می خواهیم از دیسک بخوانیم CPU مجبور است تا منتظر باشد و بیکار بماند در اینجا بهره وری منابع مطلوب نیست . در این حالت همروندی اهمیت زیادی دارد و خودی نشان می دهد . اما در صورتی که داده در رم باشد یا اندازه تراکنش کوچک باشد چنین اتفاقی نمی افتد .