

Lista zadań nr 5: transakcje i optymalizacja

T1. (2pkt.) Rozważmy następujące klasy harmonogramów (definicje poniżej): serializowalne, bez kaskadowych przerw (avoiding-cascading-aborts), odtwarzalne (recoverable) i ściśle (strict), oraz następujące protokoły 2PL, S2PL, znaczników czasowych i walidacji.

Dla każdego harmonogramu poniżej oceń do jakich klas on należy i, które z protokołów go dopuszczają. Jeśli z jakiegoś powodu nie możesz się zdecydować to uzasadnij dlaczego. Akcje, poprzedzone nazwą transakcji, są podane w kolejności w jakiej mają być wykonane, jeśli commit/abort nie jest pokazany to załóż, że może wystąpić jedynie po wszystkich podanych akcjach. Załóż, że znacznikiem czasowym transakcji T_i jest i .

1. $T1:R(X)$, $T2:R(X)$, $T1:W(X)$, $T2:W(X)$
2. $T1:W(X)$, $T2:R(Y)$, $T1:R(Y)$, $T2:R(X)$
3. $T1:R(X)$, $T2:R(Y)$, $T3:W(X)$, $T2:R(X)$, $T1:R(Y)$
4. $T1:R(X)$, $T1:R(Y)$, $T1:W(X)$, $T2:R(Y)$, $T3:W(Y)$, $T1:W(X)$, $T2:R(Y)$
5. $T1:R(X)$, $T2:W(X)$, $T1:W(X)$, $T2:Abort$, $T1:Commit$
6. $T1:R(X)$, $T2:W(X)$, $T1:W(X)$, $T2:Commit$, $T1:Commit$
7. $T1:W(X)$, $T2:R(X)$, $T1:W(X)$, $T2:Abort$, $T1:Commit$
8. $T1:W(X)$, $T2:R(X)$, $T1:W(X)$, $T2:Commit$, $T1:Commit$
9. $T1:W(X)$, $T2:R(X)$, $T1:W(X)$, $T2:Commit$, $T1:Abort$
10. $T2:R(X)$, $T3:W(X)$, $T3:Commit$, $T1:W(Y)$, $T1:Commit$,
 $T2:R(Y)$, $T2:W(Z)$, $T2:Commit$
11. $T1:R(X)$, $T2:W(X)$, $T2:Commit$, $T1:W(X)$, $T1:Commit$, $T3:R(X)$, $T3:Commit$
12. $T1:R(X)$, $T2:W(X)$, $T1:W(X)$, $T3:R(X)$, $T1:Commit$, $T2:Commit$, $T3:Commit$

Definicje. Harmonogram jest (1) *serializowalny* jeśli jest równoważny jakiemuś harmonogramowi szeregowemu; (2) *odtworzalny (recoverable)* jeśli transakcja może być zatwierdzona dopiero jeśli wszystkie transakcje, których dane przeczytała są zatwierdzone; (3) *ściśle* jeśli wszystkie dane, które czyta lub zapisuje były zmieniane wcześniej wyłącznie przez zatwierdzone transakcje. Harmonogram *unikaj kaskadowych przerw* jeśli czyta jedynie dane zapisane przez zatwierdzone transakcje.

T2. (1 pkt.) Kiedy i dlaczego transakcja jest odrzucana w przypadku stosowania metody znaczników czasowych oraz metody walidacji? Podaj moment i powód odrzucenia.

- O1. (1 pkt.)** Rozważmy operację $\text{LIMIT}(R, k)$ wybierającą próbkę k krotek z relacji R . Sformułuj i udowodnij zasady przemienności (czyli wzajemnej rozdzielności) tej operacji z pozostałymi operacjami algebry relacji: $\times, \sigma, \cup, \cap, \setminus$ oraz π , a także operacją złączenia naturalnego. Formułując zasady weź pod uwagę specyfikę operacji $\text{LIMIT}(k)$: jej zadaniem jest zwrócenie dowolnych k krotek i dwa wywołania $\text{LIMIT}(k)$ mogą być równoważne, nawet gdy zwracają różne krotki.
- O2. (2 pkt.)** Rozważmy złączenie relacji `semestr`, `przedmiot_semestr`, `przedmiot`, `grupa`, `uzytkownik (alias prac)`, `wybor`, `uzytkownik (alias stud)` z bazy danych zapisów, które posłużą nam do wybrania trójek:

`(prac.nazwisko, stud.nazwisko, semestr.nazwa)`

takich, że dany student uczęszcza na jakieś zajęcia do danego pracownika w danym semestrze. Sprawdź, w jakiej kolejności PostgreSQL wykona powyższy ciąg złączeń i czy zależy on porządku, w jakim podajesz (łączysz) relacje na liście FROM. Przedstaw plan zapytania PostgreSQL w postaci drzewa. Następnie dodaj do zapytania:

- warunek na nazwisko pracownika;
- warunek na nazwisko studenta.

Jak każdy z warunków wpływa na drzewo zapytania?