ASSIGNMENT 5 Solution

Answer 2:

1. Find the names of all experts who can repair a product with size at least 50 from the company named "Acme".

Answer:

Select extractvalue(c.factories, ‘//ProductType[@size>=50]

/Repairer[@qualification='expert'] /text()’)

From Company c

Where c.name = “Acme”;

1. List the labels of all products that have not yet been completed and have a repairer who is at least at the intermediate level from any company

Answer:

Select extractvalue(c.factories, ‘//ProductType

/Product[@completed = '' or translate(@completed, '-', '') >= translate(current-date(), '-', '')]

/../Repairer[@qualification='expert' or @qualification='intermediate']

/../Product/@label’)

From Company c;

1. List the addresses of the factories whose second machine has a code starting with "10" from companies whose name ends in "Inc"

Answer :

Select extractvalue(c.factories, ‘///Factory/Machine[2][starts-with(@code, '10')]

/../@address’)

From Company c

Where name like “%Inc”;

1. Find the average size of all products that are currently being produced.

Answer:

Select extractxml(c.factories, ‘sum(/Company/ProductType/Product[@completed = '' or translate(@completed, '-', '') >= translate(current-date(), '-', '')]/../@size)

div

count(/Company/ProductType/Product[@completed = '' or translate(@completed, '-', '') >= translate(current-date(), '-', '')]/../@size)’)

From Company c;

Ques 1. For each schedule:

1. Determine whether the schedule is view-serializable, conflict-serializable, recoverable, cascadeless and/or strict. Note: you must state separately whether each one of these holds.
2. For each of the 5 answers in the first part above, explain why or why not. In the case of the two notions of serializability, you must either give an equivalent serial schedule or a reason why the schedule is not serializable. For the other three concepts, you must give an example of a violation if the schedule does not have the property.

Answer :

The following are the schedules:

1. R3(A), R3(B), R1(C), R2(C), R2(B), R3(A), W3(A), C2, W1(C), C1, W4(B), W3(A), W4(B), C3, W4(A), W4(C), C4

Answer :

Given schedule is view-serializable

conflict-serialzable

recoverable

cascadeless

strict

Please find the conflicts below :

R3(A)-> W4(A)

R3(B)-> W4(B)

R1(C)-> W4(C)

R2(C)-> W1(C)

R2(C) -> W4(C)

R2(B) -> W4(B)

W3(A) -> W4(A)

W1(C) -> W4(C)

These give the following transaction conflicts:

T3 -> T4

T1 -> T4

T2 -> T4

T2 -> T1

1. As there is no cycle, the given schedule is conflict-serializable.
2. Each conflict-serializable is view-serializable.
3. For recoverable , if Wi(X) …Rj(X) then, Ci comes before Cj. Since given schedule satisfies this

property, Hence given schedule is recoverable

1. For cascadeless , if Wi(X) …Rj(X) then, Ci comes for Rj. Since given schedule satisfies this

property, Hence given schedule is cascadeless

1. For strict, if Wi(X) …Wj(X) then Ci comes before Cj. Since given schedule satisfies this

property, Hence given schedule is strict

1. W4(C), W3(C), W2(A), R2(B), R3(A), W4(A), W1(A), C4, C2, W1(C), R1(A), R3(B), C3, R1(B), C1

Answer :

Given schedule is view-serializable

not conflict-serialzable

recoverable

not cascadeless

not strict

Please find the conflicts below :

W4(C) -> W3(C)

W4(C) -> W1(C)

W3(C) -> W1(C)

W2(A) -> W4(A)

W2(A) -> W1(A)

W2(A) -> R3(A)

W2(A) -> R1(A)

R3(A)->W4(A)

R3(A)->W1(A)

W4(A) -> W1(A)

W4(A) -> R1(A)

These give the following transaction conflicts:

T4 -> T3

T4 -> T1

T3 -> T1

T2 -> T4

T2 -> T1

T2 -> T3

T3 -> T4

1. As there is a cycle, the given schedule is not conflict-serializable.
2. “R3(A) reads W2(A)” means C2 comes before C3 , hence we have 12 possible serial orders.

Order 1234 :

W1(A),W1(C), R1(A) ,R1(B) ,C1, W2(A), R2(B), C2 ,W3(C), R3(A),R3(B),C3,W4(C),W4(A),C4

In the original schedule W1(C) writes the final value of C

In the serial schedule W4(C) writes the final value of C.

Order 1423:

W1(A),W1(C), R1(A) ,R1(B) ,C1 ,W4(C),W4(A),C4, W2(A), R2(B), C2, W3(C), R3(A),R3(B),C3

In the original schedule W1(C) writes the final value of C

In the serial schedule W3(C) writes the final value of C.

Order 4231:

W4(C),W4(A),C4, W2(A), R2(B), C2, W3(C), R3(A),R3(B),C3, W1(A),W1(C), R1(A) ,R1(B) ,C1

In the original schedule W1(C) writes the final value of C

In the serial schedule W1(C) writes the final value of C.

All possible options of Read and Write as well as final commits are as per order in this case. Hence this schedule is “View Serializable”

1. For recoverable , if Wi(X) …Rj(X) then, Ci comes before Cj. Since given schedule satisfies this

property, Hence given schedule is recoverable

1. For cascadeless , if Wi(X) …Rj(X) then, Ci comes for Rj. In the given schedule, “R3(A) reads W2(A)” but doesn’t have C2 before R3(A) .Since given schedule doesn’t satisfy this

property, Hence given schedule is not cascadeless

1. For strict, if Wi(X) …Wj(X) then Ci comes before Cj. In the given schedule, W2(A) comes before W4(A), but C2 comes after C4, Since given schedule doesn’t satisfy this

property, Hence given schedule is not strict

1. R2(B), R4(A), R2(C), W2(B), W3(C), R1(B), W4(C), R2(A), C4, C2, R1(B), W1(A), W3(B), R3(A), C3, C1

Answer :

Given schedule is not view-serializable

not conflict-serialzable

not recoverable

not cascadeless

not strict

Please find the conflicts below :

R2(B)-> W3(B)

R4(A)-> W1(A)

R2(C)-> W3(C)

R2(C)-> W4(C)

W2(B) -> R1(B)

W2(B)->W3(B)

W3(C)->W4(C)

R2(A)->W1(A)

R1(B)->W3(B)

W1(A)->R3(A)

These give the following transaction conflicts:

T2 -> T4

T4 -> T1

T1 -> T3

T3 -> T4

T2 ->T3

T2 ->T1

1. As there is a cycle, the given schedule is not conflict-serializable.
2. Here “R1(B) reads W2(B)” so C2 comes before C1

“R3(A) reads W1(A)” so C1 comes from C3

Thus only 3 possible orders:

2134 order :

R2(B), R2(C), W2(B), R2(A),C2, R1(B), R1(B), W1(A), C1, W3(C), W3(B), R3(A) , C3, R4(A), W4(C),C4

In the original schedule, R4(A) reads A

In the serial schedule, R4(A) reads W1(A)

2143 order :

R2(B), R2(C), W2(B), R2(A),C2, R1(B), R1(B), W1(A), C1, R4(A), W4(C),C4, W3(C), W3(B), R3(A) , C3

In the original schedule, R4(A) reads A

In the serial schedule, R4(A) reads W1(A)

4213 order :

R4(A), W4(C),C4, R2(B), R2(C), W2(B), R2(A),C2, R1(B), R1(B), W1(A), C1, W3(C), W3(B), R3(A) , C3

In the original schedule, W4(C) writes the final value of C

In the serial schedule, W3(C) writes the final value of C

Hence, the given schedule is not view serializable

1. For recoverable , if Wi(X) …Rj(X) then, Ci comes before Cj. In the given schedule “R3(A) reads W1(A)” but C1 comes after C3.Hence given schedule is not recoverable.
2. For cascadeless , if Wi(X) …Rj(X) then, Ci comes for Rj. In the given schedule, “R1(B) reads W2(B)” but doesn’t have C2 before R1(B) .Since given schedule doesn’t satisfy this

property, Hence given schedule is not cascadeless

1. For strict, if Wi(X) …Wj(X) then Ci comes before Cj. In the given schedule, W3(C) comes after W4(C), but C4 comes before C3.Hence given schedule is not strict
2. R2(A), R2(B), C2, R3(B), W4(B), W1(A), R4(C), R3(A), R1(A), C1, R4(C), W4(A), W3(B), R4(C), R3(A), C3, R4(A), R4(C), C4

Answer :

Given schedule is not view-serializable

not conflict-serialzable

not recoverable

not cascadeless

not strict

Please find the conflicts below :

R2(A) -> W1(A)

R2(A) -> W4(A)

R2(B) -> W3(B)

R2(B) -> W4(B)

R3(B) -> W4(B)

W4(B) -> W3(B)

W1(A) -> W4(A)

R3(A) -> W4(A)

W4(A) -> R3(A)

W1(A) -> R3(A)

W1(A) -> R4(A)

R1(A) -> W4(A)

These give the following transaction conflicts:

T2->T1

T2->T4

T2 -> T3

T1->T4

T1->T3

T3->T4

T4->T3

1. As there is a cycle, the given schedule is not conflict-serializable.
2. Here “R3(A) reads W4(A)” so C4 comes before C3

“R3(A) reads W1(A)” so C1 comes from C3

Thus only 8 possible orders:

1423 order :

W1(A), R1(A), C1, W4(B), R4(C), R4(C), W4(A) , R4(C), R4(A), R4(C), C4, R2(A), R2(B), C2, R3(B), R3(A), W3(B), R3(A),C3

In the original schedule, R2(A) reads A

In the serial schedule, R2(A) reads W4(A)

4213 order:

W4(B), R4(C), R4(C), W4(A) , R4(C), R4(A), R4(C), C4, R2(A), R2(B), C2, W1(A), R1(A), C1, R3(B), R3(A), W3(B), R3(A),C3

In the original schedule, R2(A) reads A

In the serial schedule, R2(A) reads W4(A)

1243 order:

W1(A), R1(A), C1,R2(A),R2(B),C2, W4(B), R4(C), R4(C), W4(A) , R4(C), R4(A), R4(C), C4 ,R3(B),R3(A),W3(B),R3(A),C3

In the original schedule R2(A) read first value of schedule

In this serial schedule R2(A) reads from W1(A)

1432 -- order:

W1(A),R1(A),C1, W4(B), R4(C), R4(C), W4(A) , R4(C), R4(A), R4(C), C4, R3(B),R3(A),W3(B),R3(A),C3, R2(A),R2(B),C2

In the original schedule R2(A) read first value of schedule

In this serial schedule R2(A) reads from W1(A)

4132 -- order:

W4(B), R4(C), R4(C), W4(A) ,R4(C),R4(A),R4(C),C4,W1(A),R1(A),C1,R3(B),R3(A),W3(B),R3(A),C3,R2(A),R2(B),C2

In the original schedule R2(A) read first value of schedule

In this serial schedule R2(A) reads from W1(A)

2413 order:

R2(A),R2(B),C2,W4(B),R4(C),R4(C),R4(C),R4(A),R4(C),C4,W1(A),R1(A),C1,R3(B),R3(A),W3(B),

R3(A),C3

In the original schedule R3(B) reads B

In this serial schedule R3(B) reads W4(B)

2143 order:

R2(A),R2(B),C2,W1(A),R1(A),C1,W4(B),R4(C),R4(C),R4(C),W4(A),R4(C),R4(A),R4(C),C4,R3(B),

R3(A),W3(B),R3(A),C3

In the original schedule R3(B) reads B

In this serial schedule R3(B) reads W4(B)

4123 order:

W4(B),R4(C),R4(C),R4(C),W4(A),R4(C),R4(A),R4(C),C4,R3(B), W1(A),R1(A),C1, R2(A),R2(B),C2,

R3(B),R3(A),W3(B),R3(A),C3

In the original schedule R3(B) reads B

In this serial schedule R3(B) reads W4(B)

Hence, the given schedule is not view serializable

1. For recoverable Wi(X) --- Rj(X) then Ci comes before Cj

W4(A) – R3(A) --- C4 should come before C3 but this is not happening hence its not recoverable.

1. For cascadeless Wi(X) – Rj(X) then Ci comes before Rj

Since W4(A) –> R3(A) without C4 between the transactions. Hence it is not cascadeless .

1. For strict Wi(X) --- Wj(X) then Ci comes before Cj

W4(B) –> W3(B) hence C4 should come before C3 but this is not happening hence it is not strict.