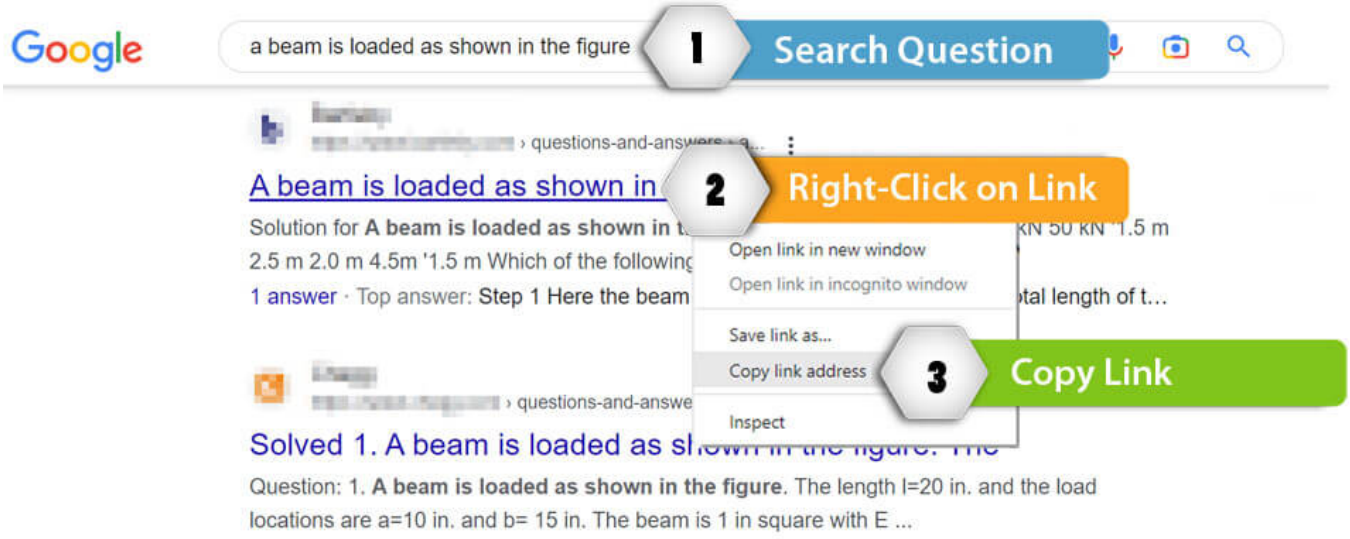


Answer

a)

Let the 4 transactions be named as T1, T2, T3 and T4.

The given schedule in tabular form is :

Timestamp	T1	T2	T3	T4
1	R1(A)			
2				R4(A)
3	W1(A)			
4			W3(B)	
5		R2(A)		
6		R2(B)		

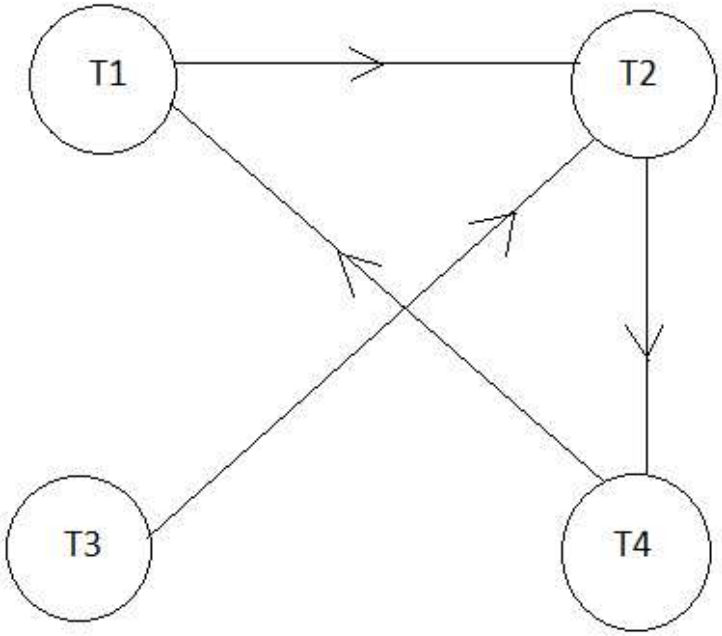
7		W2(C)		
8				R4(B)
9				R4(C)
10		R2(D)		
11			R3(E)	
12	C1			
13		C2		
14			C3	
15				C4

For the precedence graph, there will be 4 vertices namely, T1, T2, T3 and T4.

From the above table, the following points are inferred to draw the precedence graph :

1. At timestamps 2 and 3, there is a read write conflict on data item A from T4 to T1 and so there will be an edge from T4 to T1.
2. At timestamps 3 and 5, there is a write read conflict on data item A from T1 to T2 and so there will be an edge from T1 to T2.
3. At timestamps 4 and 6, there is a write read conflict on data item B from T3 to T2 and so there will be an edge from T3 to T2.
4. At timestamps 7 and 9, there is a write read conflict on data item C from T2 to T4 and so there will be an edge from T2 to T4.

The precedence graph is :



In the above precedence graph, there is a cycle T4 - T1 - T2 - T4.

Hence, it is not conflict serializable.

b)

Let the 4 transactions be named as T1, T2, T3 and T4.

The given schedule in tabular form is :

Timestamp	T1	T2	T3	T4
1	R1(A)			
2	W1(A)			
3		R2(A)		

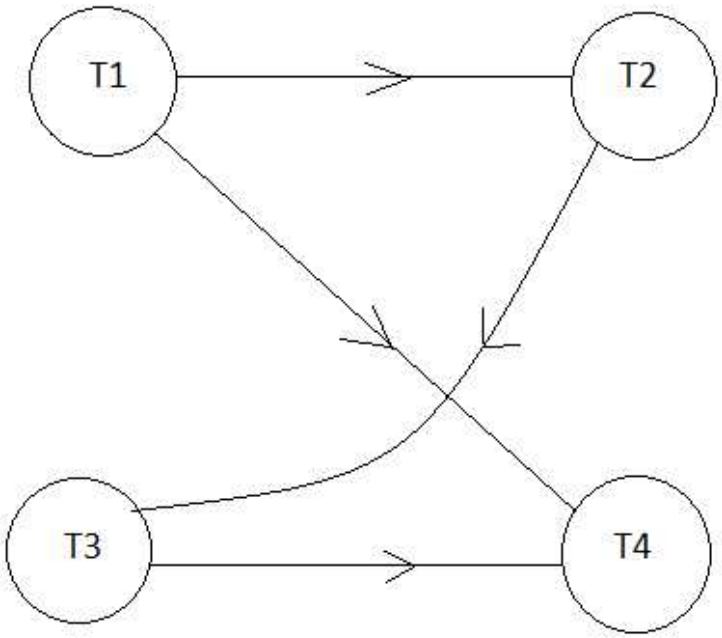
4		R2(B)		
5			W3(B)	
6		W2(C)		
7				R4(A)
8				R4(B)
9		R2(D)		
10			R3(E)	
11	C1			
12		C2		
13			C3	
14				C4

For the precedence graph, there will be 4 vertices namely, T1, T2, T3 and T4.

From the above table, the following points are inferred to draw the precedence graph :

1. At timestamps 2 and 3, there is a write read conflict on data item A from T1 to T2 and so there will be an edge from T1 to T2.
2. At timestamps 2 and 7, there is a write read conflict on data item A from T1 to T4 and so there will be an edge from T1 to T4.
3. At timestamps 4 and 5, there is a read write conflict on data item B from T2 to T3 and so there will be an edge from T2 to T3.
4. At timestamps 5 and 8, there is a read write conflict on data item B from T3 to T4 and so there will be an edge from T3 to T4.

The precedence graph is :



The above precedence graph does not contain any cycle and hence, it is conflict serializable.

An equivalent serial schedule is obtained by performing topological sort on the precedence graph as follows :

Step 1 :

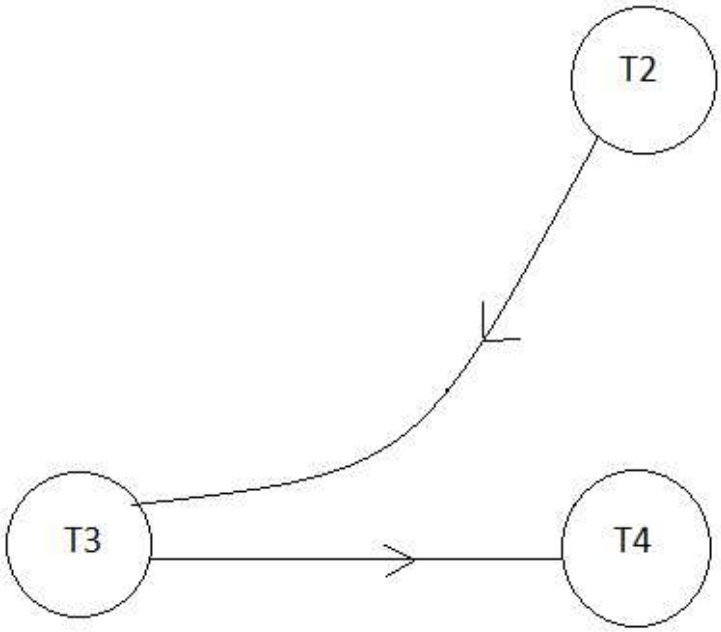
Select the vertex having 0 indegree from the precedence graph.

The vertex selected is T1.

Remove the vertex and its adjacent edges.

So, the first vertex in the topological sorting is T1.

The precedence graph till now is :



Step 2 :

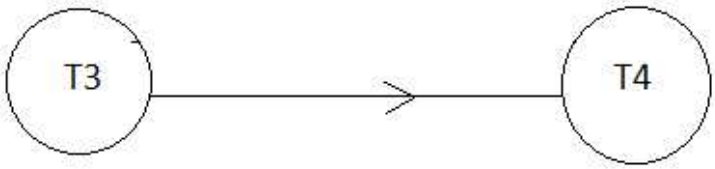
Select the vertex having 0 indegree from the precedence graph obtained in previous step.

The vertex selected is T2.

Remove the vertex and its adjacent edges.

So, the next vertex in the topological sorting is T2.

The precedence graph till now is :



Step 3 :

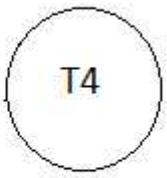
Select the vertex having 0 indegree from the precedence graph obtained in previous step.

The vertex selected is T3.

Remove the vertex and its adjacent edges.

So, the next vertex in the topological sorting is T3.

The precedence graph till now is :



Step 5 :

The remaining vertex is T4.

So, the equivalent serial schedule is T1 - T2 - T3 - T4.

Hence, the equivalent serial schedule is :

Timestamp	T1	T2	T3	T4
1	R1(A)			
2	W1(A)			
3		R2(A)		
4		R2(B)		
5		W2(C)		
6		R2(D)		
7			W3(B)	
8			R3(E)	
9				R4(A)
10				R4(B)
11	C1			
12		C2		
13			C3	
14				C4

Likes: 0 **Dislikes: 0**