

Βάσεις Δεδομένων 2

Φροντιστήριο
Δοσοληψίες
Τεχνικές ελέγχου συνδρομικότητας

Ημερ: 27/4/2010
Ακ. Έτος 2009-10

Δοσοληψία-ορισμός

- Είναι μια λογική μονάδα επεξεργασίας της βάσης δεδομένων που περιλαμβάνει μια ή περισσότερες πράξεις προσπέλασης στη βάση δεδομένων. Αυτές μπορεί να περιλαμβάνουν πράξεις εισαγωγής, διαγραφής, τροποποίησης ή ανάκτησης.
- **Το πρόβλημα της απώλειας των ενημερώσεων**
 - εμφανίζεται όταν οι πράξεις δυο δοσοληψιών, που προσπελαίνουν τα ίδια στοιχειώδη δεδομένα, διαπλέκονται κατά τρόπο που καθιστά την τιμή κάποιου στοιχειώδους δεδομένου λανθασμένη.
- **Το πρόβλημα της προσωρινής ενημέρωσης**
 - εμφανίζεται όταν μια δοσοληψία ενημερώνει ένα στοιχείο της βάσης δεδομένων και στη συνέχεια αποτυγχάνει για κάποιο λόγο. Μια άλλη δοσοληψία προσπελαίνει το στοιχείο που έχει ενημερωθεί, πριν αυτό επανέλθει στην αρχική του τιμή.(dirty data)
- **Το πρόβλημα της λανθασμένης συνάθροισης**
 - αν μια δοσοληψία υπολογίζει μια συναθροιστική συνάρτηση σ' ένα πλήθος εγγραφών ενόσω άλλες δοσοληψίες ενημερώνουν κάποιες από αυτές τις εγγραφές, η συναθροιστική συνάρτηση μπορεί να χρησιμοποιήσει κάποιες τιμές πριν ενημερωθούν και άλλες αφού ενημερωθούν.

Χρονοπρόγραμμα(ή χρονικό)

S των n δοσοληψιών T_1, T_2, \dots, T_n είναι μια διάταξη των πράξεων των δοσοληψιών η οποία υπόκειται στον περιορισμό ότι, για κάθε δοσοληψία T_i που συμμετέχει στο S , οι πράξεις της T_i στο S πρέπει να εμφανιστούν με την ίδια σειρά που εμφανίζονται στην T_i .

- **Σειριακό χρονοπρόγραμμα**

- για κάθε δοσοληψία T που συμμετέχει στο χρονοπρόγραμμα, όλες οι πράξεις της T να εκτελούνται διαδοχικά στο χρονοπρόγραμμα

- **Σειριοποιήσιμο χρονοπρόγραμμα**

- αν είναι ισοδύναμο με κάποιο σειριακό

Ισοδυναμία Χρονοπρογραμμάτων

- **Βάση αποτελέσματος**

- Παράγουν την ίδια τελική κατάσταση της βάσης δεδομένων

- **Βάση Αντιθέσεων**

Αν η διάταξη κάθε ζεύγους αντιμεταθεμένων πράξεων είναι η ίδια και στα δύο χρονοπρογράμματα.

- Δύο πράξεις s' ένα χρονοπρόγραμμα είναι σε **αντίθεση** αν
 - ανήκουν σε διαφορετικές δοσοληψίες
 - προσπελαύνουν το ίδιο στοιχείο x
 - τουλάχιστον μια από τις 2 είναι η εγγραφή

- **Βάση όψεων**

- Δύο χρονοπρογράμματα καλούνται **ισοδύναμα όψεων** αν ισχύουν οι εξής συνθήκες :
 - Το ίδιο σύνολο δοσοληψιών συμμετέχει και στο χρονοπρόγραμμα S και στο S' και τα S και S' περιλαμβάνουν τις ίδιες πράξεις των συγκεκριμένων δοσοληψιών
 - Για κάθε πράξη $r_i(X)$ της T_i στο S , αν η τιμή της X που διαβάζεται από την πράξη έχει γραφτεί από μια πράξη $w_i(X)$ της T_k (ή αν είναι η αρχική τιμή του X πριν την έναρξη του χρονοπρογράμματος), η ίδια συνθήκη πρέπει να ισχύει για την τιμή του X που διαβάζεται από την πράξη $r_i(X)$ της T_i στο S' .
 - Αν η πράξη $w_k(Y)$ της T_k είναι η τελευταία πράξη που τροποποιεί το στοιχείο Y στο S , τότε η $w_k(Y)$ της T_k πρέπει να είναι η τελευταία πράξη που τροποποιεί το Y στο S' .

Γράφος προτεραιότητας

- Κατευθυνόμενος γράφος $G=(N,E)$
 - Σύνολο κόμβων $N=\{T_1, T_2, \dots, T_n\}$
 - Σύνολο ακμών $E=\{\alpha_1, \alpha_2, \dots, \alpha_m\}$
- Κάθε ακμή α_i στον γράφο έχει την μορφή $(T_j \rightarrow T_k)$
 - Μια τέτοια ακμή δημιουργείται στον γράφο, αν μία από τις πράξεις στο T_j εμφανίζεται στο χρονοπρόγραμμα πριν από μία πράξη σε αντίθεση στο T_k .
- **Εάν ο γράφος δεν περιέχει κύκλο το χρονοπρόγραμμα είναι σειριοποιήσιμο.**

Αλγόριθμος ελέγχου σειριοποιησιμότητας αντιθέσεων

1. Για κάθε δοσοληψία T_i που συμμετέχει στο χρονοπρόγραμμα S , δημιούργησε έναν κόμβο T_i στο γράφο προήγησης
2. Για κάθε περίπτωση στο S όπου η T_j εκτελεί μία πράξη **ανάγνωσης_στοιχείου(X)** μετά από μία εντολή **εγγραφή_στοιχείου(X)** που εκτελείται από την T_i δημιούργησε μία ακμή ($T_i \rightarrow T_j$) στον γράφο προήγησης.
3. Για κάθε περίπτωση στο S όπου η T_j εκτελεί μία πράξη **εγγραφή_στοιχείου(X)** μετά από μία εντολή **ανάγνωσης_στοιχείου(X)** που εκτελείται από την T_i δημιούργησε μία ακμή ($T_i \rightarrow T_j$) στον γράφο προήγησης.
4. Για κάθε περίπτωση στο S όπου η T_j εκτελεί μία πράξη **εγγραφή_στοιχείου(X)** αφού η T_i εκτελέσει μία εντολή **εγγραφή_στοιχείου(X)**, δημιούργησε μία ακμή ($T_i \rightarrow T_j$) στον γράφο προήγησης.
5. Το χρονοπρόγραμμα S είναι σειριοποιήσιμο αν και μόνον αν ο γράφος προήγησης δεν έχει κυκλώματα

Ασκήσεις

Άσκηση 1

- Υποθέστε τα χρονοπρογράμματα T1 και T2
- Υποθέστε ότι η συνθήκη συνέπειας της βάσης δεδομένων είναι $(A=0 \text{ or } B=0)$ και (β) οι αρχικές τιμές των A και B είναι 0 ($A=B=0$).

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)

■ Ζητούμενα:

1. Δείξτε ότι κάθε σειριακή εκτέλεση των δυο χρονοπρογραμμάτων διατηρεί την συνέπεια της βάσης δεδομένων
2. Αναφέρετε μία μη σειριακή (συνδρομική) εκτέλεση των T1 και T2 η οποία παράγει ένα μη σειριοποιήσιμο χρονοπρόγραμμα.
3. Υπάρχει συνδρομική εκτέλεση των T1 και T2 από την οποία να παράγεται ένα σειριοποιήσιμο χρονοπρόγραμμα;

Άσκηση 1 – Λύση – ερώτημα 1

1. Δείξτε ότι κάθε σειριακή εκτέλεση των δυο χρονοπρογραμμάτων διατηρεί την συνέπεια της βάσης δεδομένων
- $T1 \rightarrow T2$
 - Εκτελείται η T1 :A=0 , B=1
 - Εκτελείται η T2 :A=0 , B=1
 - Η βάση σε συνεπή κατάσταση γιατί A=0
 - $T2 \rightarrow T1$
 - Εκτελείται η T2 :A=1 , B=0
 - Εκτελείται η T1 :A=1 , B=0
 - Η βάση σε συνεπή κατάσταση γιατί B=0

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)

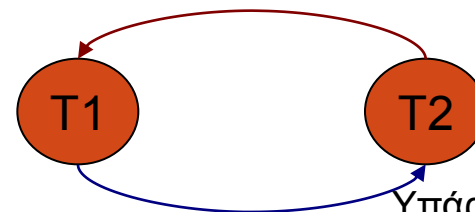
Άσκηση 1 – Λύση – ερώτημα 2

2. Αναφέρετε μία μη σειριακή (συνδρομική) εκτέλεση των T1 και T2 η οποία παράγει ένα μη σειριοποιήσιμο χρονοπρόγραμμα.

• Λύση:

Read ₁ (A)	←
Read ₁ (B)	
Read ₂ (A)	
Read ₂ (B)	←
if ₁ A = 0 then B:=B+1;	
Write ₁ (B)	←
if ₂ B = 0 then A:=A+1;	
Write ₂ (A)	←

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)



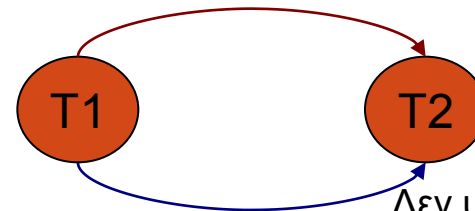
Υπάρχει κύκλος → Το πρόγραμμα είναι μη σειριοποιήσιμο

Άσκηση 1 – Λύση – ερώτημα 3

3. Υπάρχει συνδρομική εκτέλεση των T1 και T2 η οποία να παράγει ένα σειριοποιήσιμο χρονοπρόγραμμα;
- Λύση:

Read ₁ (A)	←
Read ₁ (B)	
Read ₂ (A)	
if ₁ A = 0 then B:=B+1;	
Write ₁ (B)	←
Read ₂ (B)	←
if ₂ B = 0 then A:=A+1;	
Write ₂ (A)	←

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)



Δεν υπάρχει κύκλος →
Το πρόγραμμα είναι
σειριοποιήσιμο

Άσκηση 2

- Έστω οι ακόλουθες 2 δοσοληψίες
 1. Προσθέστε εντολές κλειδώματος και ξεκλειδώματος ώστε οι δύο δοσοληψίες να ακολουθούν το πρωτόκολλο κλειδώματος δύο φάσεων
 2. Στις δοσοληψίες που προέκυψαν από το ερώτημα (1), υπάρχει περίπτωση αδιεξόδου (deadlock);

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)

Άσκηση 2 – Λύση – ερώτημα 1

1. Προσθέστε εντολές κλειδώματος και ξεκλειδώματος ώστε οι δύο δοσοληψίες να ακολουθούν το πρωτόκολλο κλειδώματος δύο φάσεων
- Λύση:

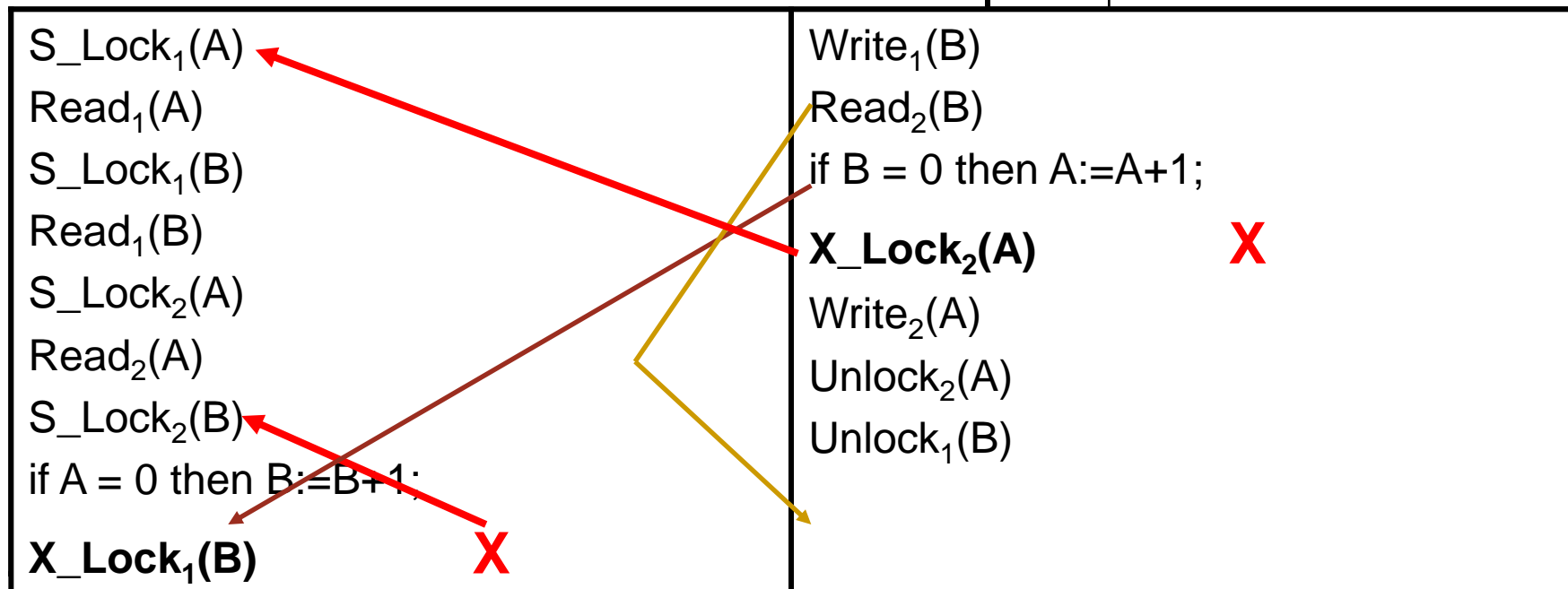
T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)

S_Lock ₁ (A) Read ₁ (A) S_Lock ₁ (B) Read ₁ (B) S_Lock ₂ (A) Read ₂ (A) if A = 0 then B:=B+1; X_Lock ₁ (B) Write1(B)	Unlock1(B) Unlock1(A) S_Lock ₂ (B) Read2(B) if B = 0 then A:=A+1; X_Lock2(A) Write2(A) Unlock2(A) Unlock2(B)
---	---

Άσκηση 2 – Λύση – ερώτημα 2

- Στις δοσοληψίες που προέκυψαν από το ερώτημα (1), υπάρχει περίπτωση αδιεξόδου (deadlock);
- **Λύση:** Περίπτωση που υπάρχει deadlock

T1	Read(A) Read(B) if A = 0 then B:=B+1; Write(B)
T2	Read(A) Read(B) if B = 0 then A:=A+1; Write(A)



Άσκηση 3

Στα παρακάτω $R_i(X)$ σημαίνει ότι η δοσοληψία i διαβάζει το δεδομένο X και $W_i(X)$ σημαίνει ότι η δοσοληψία i γράφει το δεδομένο X .

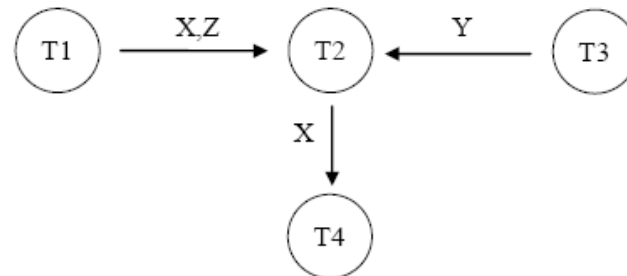
Για κάθε ένα από τα παρακάτω προγράμματα (schedules) απαντήστε τις παρακάτω ερωτήσεις:

1. Είναι το πρόγραμμα σειριοποιήσιμο; Αν ναι, δώστε **όλα** τα ισοδύναμα σειριακά προγράμματα. Αν όχι, εξηγήστε σύντομα γιατί.
 2. Μπορεί να προκύψει το πρόγραμμα αυτό από πρωτόκολλο συντονισμού με κλείδωμα σε δυο φάσεις (two-phase locking); Σύντομα εξηγήστε γιατί ναι ή γιατί όχι.
- **Πρόγραμμα 1** $R_1(X) R_2(X) R_3(Y) W_2(X) W_1(Z) W_2(Y) R_4(X) W_2(Z)$
 - **Πρόγραμμα 2** $R_1(X) R_1(Z) W_2(X) R_3(Y) R_2(Y) R_3(X) W_3(Z) W_1(Y) R_2(Z)$
 - **Πρόγραμμα 3** $W_1(Z) R_3(Y) R_2(Z) W_3(X) R_3(Z) W_4(X) W_2(Y) R_4(Y)$

Λύση Άσκησης 3 - Ερώτημα 1

Πρόγραμμα 1 R1(X) R2(X) R3(Y) W2(X) W1(Z) W2(Y) R4(X) W2(Z)

- Σχεδιάζουμε τον γράφο σειριοποιησιμότητας

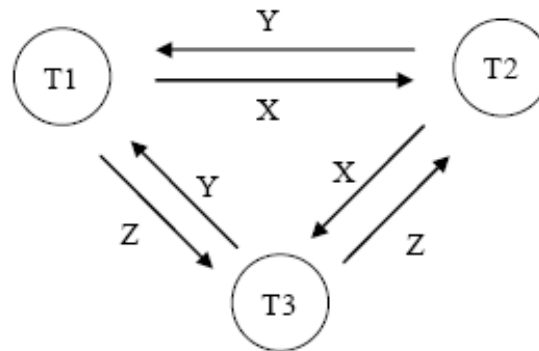


- Στον παραπάνω γράφο δεν υπάρχει κανένας κύκλος συνεπώς το αντίστοιχο πρόγραμμα είναι σειριοποιήσιμο.
- Τα ισοδύναμα σειριακά προγράμματα είναι τα:
 1. $T1 \rightarrow T3 \rightarrow T2 \rightarrow T4$
 2. $T3 \rightarrow T1 \rightarrow T2 \rightarrow T4$.

Λύση Άσκησης 3 - Ερώτημα 1

Πρόγραμμα 2 R1(X) R1(Z) W2(X) R3(Y) R2(Y) R3(X) W3(Z) W1(Y) R2(Z)

- Για το δεύτερο πρόγραμμα ο αντίστοιχος γράφος σειριοποιησιμότητας είναι ο παρακάτω:

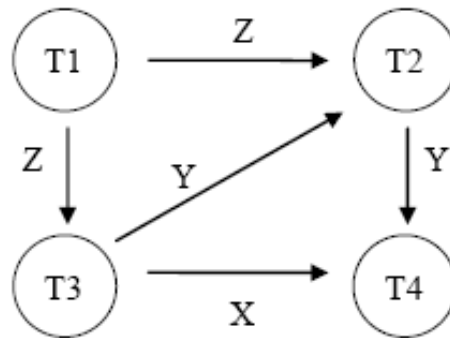


- Εφόσον σχηματίζονται
 1. $T1 \rightarrow T2 \rightarrow T3 \rightarrow T1$,
 2. $T1 \leftrightarrow T2$,
 3. $T1 \leftrightarrow T3$,
 4. $T2 \leftrightarrow T3$ και
 5. $T1 \rightarrow T3 \rightarrow T2 \rightarrow T1$,το πρόγραμμα δεν είναι σειριοποιήσιμο.

Λύση Άσκησης 3 - Ερώτημα 1

■ Πρόγραμμα 3 $W1(Z) R3(Y) R2(Z) W3(X) R3(Z) W4(X) W2(Y) R4(Y)$

- Τέλος για το τρίτο πρόγραμμα ο γράφος σειριοποιησιμότητας φαίνεται στο επόμενο σχήμα.



- Στον παραπάνω γράφο δεν υπάρχει κανένας κύκλος συνεπώς το αντίστοιχο πρόγραμμα είναι σειριοποιήσιμο.
- Το ισοδύναμο σειριακό πρόγραμμα είναι το $T1 - T3 - T2 - T4$.

Λύση Άσκησης 3 - Ερώτημα 2

Πρόγραμμα 1 R1(X) R2(X) R3(Y) W2(X) W1(Z) W2(Y) R4(X) W2(Z)

Πρόγραμμα 2 R1(X) R1(Z) W2(X) R3(Y) R2(Y) R3(X) W3(Z) W1(Y) R2(Z)

Πρόγραμμα 3 W1(Z) R3(Y) R2(Z) W3(X) R3(Z) W4(X) W2(Y) R4(Y)

- Το πρόγραμμα 2 δεν είναι δυνατόν να προκύψει από το πρωτόκολλο συντονισμού με κλείδωμα σε δυο φάσεις (two-phase locking) διότι δεν είναι σειριοποιήσιμο. Αντίθετα το πρώτο και το τρίτο πρόγραμμα μπορούν να προκύψουν από το πρωτόκολλο συντονισμού με κλείδωμα σε δυο φάσεις με τον ακόλουθο τρόπο:

- *Πρώτο πρόγραμμα:*

**S1(X) R1(X) X1(Z) U1(X) X2(X) R2(X) S3(Y) R3(Y) U3(Y) C3 W2(X) W1(Z)
U1(Z) C1 X2(Y) W2(Y) X2(Z) U2(X) S4(X) R4(X) U4(X) C4 W2(Z) U2(Y,Z) C2**

- *Τρίτο πρόγραμμα:*

**X1(Z) W1(Z) U1(Z) C1 S3(Y) R3(Y) S2(Z) R2(Z) X3(X) W3(X) S3(Z) R3(Z)
U3(X,Y,Z) C3 X4(X) W4(X) X2(Y) W2(Y) U2(Y,Z) C2 S4(Y) R4(Y) U4(X,Y) C4**

Άσκηση 4

Ας υποθέσουμε ότι έχουμε πέντε δοσοληψίες που κάνουν τα ακόλουθα:

Δοσοληψία 1 R1(Z) R1(X) W1(U)

Δοσοληψία 2 R2(X) W2(X)

Δοσοληψία 3 R3(X) R3(Y) W3(Z)

Δοσοληψία 4 R4(Z) R4(X) R4(Y)

Δοσοληψία 5 W5(U) W5(Y) R5(Y)

και σε μια εκτέλεση τους παράγεται το Πρόγραμμα 1 ενώ σε μία άλλη το Πρόγραμμα 2.

Πρόγρ. 1 R2(X) W5(U) R1(Z) W2(X) R1(X) W5(Y) R3(X) R4(Z) R3(Y) R4(X) R5(Y) W3(Z) R4(Y) W1(U)

Πρόγρ. 2 R3(X) R1(Z) W5(U) R1(X) R4(Z) R2(X) R3(Y) W2(X) W5(Y) W1(U) R4(X) R5(Y) W3(Z) R4(Y)

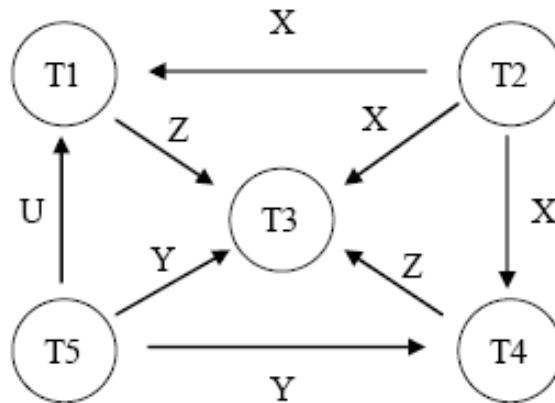
Για κάθε ένα από τα παραπάνω προγράμματα

- (α) αποδείξτε ότι είναι ή δεν είναι σειριοποιήσιμο,
- (β) σε περίπτωση που είναι, δώστε **όλα** τα ισοδύναμα σειριακά προγράμματα, και
- (γ) σε περίπτωση που δεν είναι, δώστε ένα λίγο αλλαγμένο πρόγραμμα που θα το έκανε σειριοποιήσιμο. Το νέο πρόγραμμα θα πρέπει να είναι ένα έγκυρο χρονοπρόγραμμα, το οποίο να προκύπτει από τις δοσοληψίες που δίνονται στην εκφώνηση. Δηλαδή, δεν μπορείτε να προσθέσετε ή να αφαιρέσετε πράξεις (εγγραφής, ανάγνωσης) ή να αλλάξετε τη σειρά των πράξεων σε κάθε δοσοληψία.

Λύση Άσκησης 4

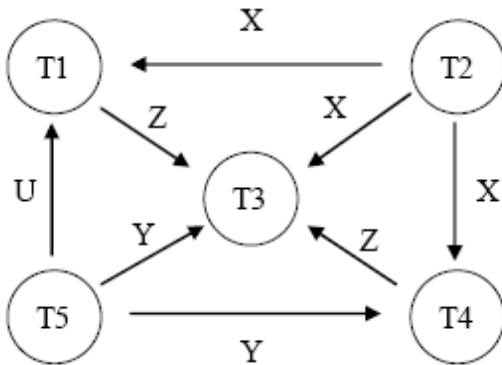
Πρόγρ. 1 R2(X) W5(U) R1(Z) W2(X) R1(X) W5(Y) R3(X) R4(Z)
R3(Y) R4(X) R5(Y) W3(Z) R4(Y) W1(U)

- Γράφος σειριοποιησιμότητας:



- Εφόσον ο γράφος δεν έχει κύκλους, το πρώτο πρόγραμμα είναι σειριοποιήσιμο.

Λύση Άσκησης 4

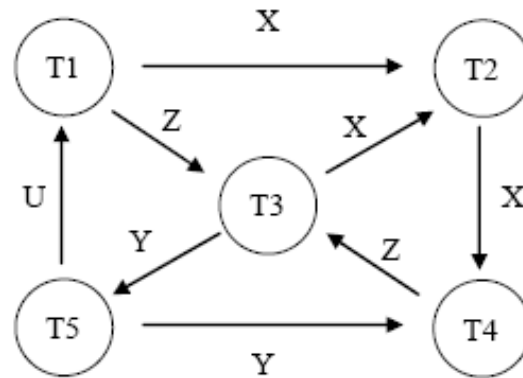


- Εφόσον ο γράφος για το πρώτο πρόγραμμα δεν έχει κύκλους, αυτό είναι σειριοποιήσιμο και τα ισοδύναμα σειριακά πρόγραμμα, έτσι όπως προκύπτουν από την τοπολογική ταξινόμηση του γράφου σειριοποιησιμότητας, είναι τα:

1. T2 – T5 – T1 – T4 – T3
2. T2 – T5 – T4 – T1 – T3
3. T5 – T2 – T1 – T4 – T3
4. T5 – T2 – T4 – T1 – T3

Λύση Άσκησης 4

- *Πρόγρ. 2* R3(X) R1(Z) W5(U) R1(X) R4(Z) R2(X) R3(Y) W2(X) W5(Y) W1(U) R4(X) R5(Y) W3(Z) R4(Y)
- Γράφος σειριοποιησιμότητας:



- Το δεύτερο πρόγραμμα δεν είναι σειριοποιήσιμο διότι ο αντίστοιχος γράφος σειριοποιησιμότητας περιέχει τέσσερις κύκλους:
 1. $T1 \rightarrow T3 \rightarrow T5 \rightarrow T1$,
 2. $T3 \rightarrow T2 \rightarrow T4 \rightarrow T3$,
 3. $T3 \rightarrow T5 \rightarrow T4 \rightarrow T3$,
 4. $T1 \rightarrow T2 \rightarrow T4 \rightarrow T3 \rightarrow T5 \rightarrow T1$.

Λύση Άσκησης 4

Το χρονοπρόγραμμα μπορεί να γίνει σειριοποιήσιμο με τις παρακάτω αλλαγές:

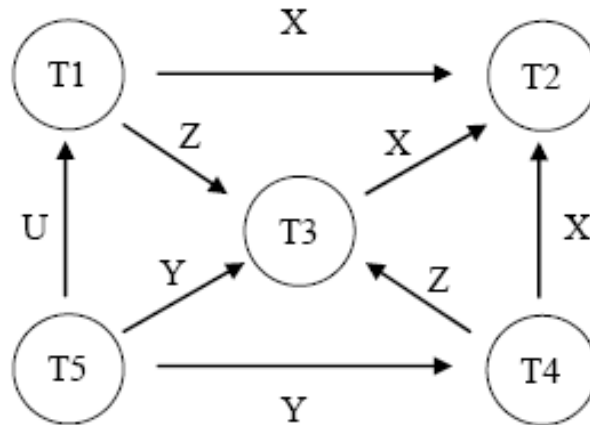
- *Πρόγρ. 2* R3(X) R1(Z) W5(U) R1(X) R4(Z) R2(X) **R3(Y) W2(X)**
W5(Y) W1(U) R4(X) R5(Y) W3(Z) R4(Y)



- *Πρόγρ. 2* R3(X) R1(Z) W5(U) R1(X) R4(Z) R2(X) ~~R3(Y) W2(X)~~
W5(Y) **R3(Y)** W1(U) R4(X) R5(Y) W3(Z) R4(Y) **W2(X)**

Λύση Άσκησης 4

- *Πρόγρ. 2* R3(X) R1(Z) W5(U) R1(X) R4(Z) R2(X) W5(Y) R3(Y)
W1(U) R4(X) R5(Y) W3(Z) R4(Y) W2(X)



- Τα ισοδύναμα σειριακά προγράμματα είναι τα:
 1. T5 – T1 – T4 – T3 – T2 και
 2. T5 – T4 – T1 – T3 – T2.