

**Κατανεμημένα Συστήματα**  
**Προπτυχιακό Μάθημα Επιλογής**  
**Εαρινό Εξάμηνο Ακ. Έτους 2006-07**  
**Παναγιώτα Φατούρου**

**2<sup>ο</sup> Σετ Ασκήσεων**

*Προθεσμία Παράδοσης: Πέμπτη, 21/6/2007, στη θυρίδα μου*

**Άσκηση 1 (Mutual Exclusion)**

- α. Παρουσιάστε σενάριο εκτέλεσης του ανιχνευτή συμφόρησης (contention detector, σελ. 24 σετ διαφανειών Mutual Exclusion) στο οποίο καμία διεργασία δεν επιστρέφει win.
- β. Αποδείξτε ότι ο μη-συμμετρικός αλγόριθμος αμοιβαίου αποκλεισμού για 2 διεργασίες (σελ. 14 σετ διαφανειών Mutual Exclusion) επιτυγχάνει αμοιβαίο αποκλεισμό και αποφυγή αδιεξόδου.
- γ. Είναι σωστός ο αλγόριθμος Bakery αν παραληφθούν οι  $choosing[j]$ ,  $1 \leq j \leq n$ , μεταβλητές και όλες οι γραμμές κώδικα που αναφέρονται σε αυτόν; Γιατί ναι, γιατί όχι;
- δ. Παρουσιάστε μια εκτέλεση του αλγορίθμου Bakery στην οποία οι τιμές που αποθηκεύονται στις  $number[j]$ ,  $1 \leq j \leq n$ , μεταβλητές είναι μη-πεπερασμένες στο πλήθος.
- ε. Η εταιρεία Lucifer Technologies Inc. έχει πρόσφατα εμφανίσει στην αγορά το νέο της πολυ-επεξεργαστή. Δυστυχώς, αυτή η μηχανή παρουσιάζει ένα περίεργο πρόβλημα: Αν περισσότερες από δύο διεργασίες προσβούν οποιαδήποτε διαμοιραζόμενη μεταβλητή ταυτόχρονα, η μηχανή καταρρέει (crashes) και πιάνει φωτιά. Η μηχανή έχει ήδη κυκλοφορήσει στην αγορά έτσι είναι δύσκολο να αποσυρθεί. Παρουσιάστε πρωτόκολλο αμοιβαίου αποκλεισμού, το οποίο δεν θα οδηγήει στην κατάρρευση της μηχανής.

**Υπόδειξη:** Μήπως κάποιο/α από τα πρωτόκολλα που έχετε ήδη διδαχθεί είναι κατάλληλο/α;

**Άσκηση 2 (Consensus)**

Το πρόβλημα k-set consensus ορίζεται ως εξής. Κάθε επεξεργαστής ξεκινά με κάποια αυθαίρετη ακέραια τιμή  $x_i$  και πρέπει να επιστρέψει μια ακέραια τιμή  $y_i$  έτσι ώστε:

**Validity:**  $y_i \in \{x_0, \dots, x_{n-1}\}$

**k-Agreement:** ο αριθμός των διαφορετικών τιμών που επιστρέφουν οι διεργασίες είναι το πολύ k.

Αποδείξτε ότι ο αλγόριθμος που περιγράφεται στο παρακάτω σχήμα επιλύει το πρόβλημα  $k$ -set consensus δεδομένου ότι μπορούν να συμβούν  $f$  crash failures στο σύστημα, όπου  $f < n$ . Ποια είναι η χρονική πολυπλοκότητα του αλγορίθμου αυτού;

*Initially*  $V = \{x_i\}$

**Round  $r$ ,  $0 < r \leq f/k + 1$ :**

send  $V$  to all processors;

receive  $S_j$  from  $p_j$ ,  $0 \leq j \leq n-1$ ,  $j \neq i$ ;

$V := V \cup \bigcup_{i=0}^{n-1} S_j$

if ( $r == f/k+1$ ) then  $y := \min(V)$ ;