ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ 2016-2017

ΑΝΑΦΟΡΑ ΕΡΓΑΣΤΗΡΙΑΚΗΣ ΑΣΚΗΣΗΣ 3

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΤΟΥΡΛΙΔΑ ΒΑΓΙΑ

ΑΜ : 6233

α) multiproc 1

Οι διεργασίες γενικά τρέχουν ανεξάρτητα χωρίς να επιρεάζουν η μία την άλλη όμως στην ασκησή μας ζητείται όλες οι διεργασίες να καλούν την display() πράγμα το οποίο σημαίνει ότι είναι πολύ πιθανό παραπάνω από μία διεργασίες να προσπαθήσουν να την εκτελέσουν ταυτόχρονα γιαυτό ακριβώς το λόγο όταν προσπαθούμε να εκτελέσουμε για πρώτη φορά το πρόγραμμα χωρίς semaphores τα αποτελέσματα μας είναι μπερδεμένα.

Για να επιλύσουμε λοιπόν το πρόβλημα αυτό κάναμε χρήση σηματοφόρων με την βοήθεια των οποίων συγχρονίσαμε της διεργασίες δηλαδή όταν μια διεργασία μπαίνει στην κρίσιμη περιοχή(στην περίπτωση μας η εκτέλεση της display)όλες οι άλλες διεργασίες μπλοκάρονται μέχρι η διεργασία αυτή να τελιώσει με την κρίσιμη περιοχή.Ο τρόπος με τον οποίο μπλοκάρουμε τις διεργασίες μας είναι με την τεχνική του busy wait μια μέθοδο με την οποία κάνουμε μια διεργασία να περιμένει.

b) multiproc 2

Στο δεύτερο ερώτημα θέλαμε να εκτελούμε δυο διαφορετικές συναρτήσεις και λειτουργήσαμε όπως και στο πρώτο ερώτημα δηλαδή κάναμε χρήση σηματοφόρων για να εξασφαλίσουμε πάλι ότι μόνο μια διεργασία κάλει την συνάρτηση κάθε φορά στη συνέχεια κάναμε ξανά χρήση της τεχνικής busy wait μετρώντας κάθε φορά πόσες διεργασίες έχουν εκτελέσει την init() και συγκρίνοντας με τον συνολικό αριθμό των διεργασιών που έπρεπε να την εκτελέσουν μπορούσαμε να μπλοκάρουμε τις διεργασίες στο σημείο αυτό μέχρι όλες να έχουν εκτελέσει την init() και στη συνέχεια να δράσουν όπως και στο ερώτημα α.

c)multithread 1

Τα thread έχουν κοινή global memory γιαυτό το λόγο είναι πολύ εύκολο να δημιουργηθούν race conditions.Επείδη τα thread τρέχουν παράλληλα όταν προσπαθούμε να καλέσουμε την display() μέσα στην συνάρτηση που καλούν τα thread όταν αρχικοποιούνται τα αποτελέσματα που μας εκτυπώνονται είναι μπερδεμένα.Για να ξεπεράσουμε αυτό το εμπόδιο κάναμε χρήση των pthread_mutex με την βοήθεια των οποίων εξασφαλίσαμε πως όταν ένα thread τρέχει την display() όλα τα άλλα θα περιμένουν.

d)multithread 2

Επειδή όπως αναφέραμε και παραπάνω τα thread τρέχουν παράλληλα όταν προσπαθούμε να καλέσουμε την init() μέσα στην συνάρτηση που καλούν τα thread όταν αρχικοποιούνται τα αποτελέσματα που μας εκτυπώνονται είναι μπερδεμένα γιαύτο το λόγο κάνουμε πάλι χρήση σεμαφόρων όμοια με πριν.Στη συνέχεια για να εξασφαλίσομε ότι όλα τα thread θα εκτελέσουν πρώτα την init() κάνουμε πάλι χρήση της τεχνικής busy wait με παρόμοια λογική όπως και στο ερώτημα b.