ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

ΑΝΑΦΟΡΑ 3ης ΑΣΚΗΣΗΣ

ΧΑΡΑΛΑΜΠΟΠΟΥΛΟΣ ΠΑΝΑΓΙΩΤΗΣ

AM:5681

E-MAIL: charalabop@ceid.upatras.gr

1)Για να αποφυγουμε το ανακατεμα των μηνυματων που τυπωνονται γινεται χρηση σημαφορων με τους οποιους υλοποιειται αμοιβαιος αποκλεισμος στα κρισιμα σημεια του κωδικα δηλαδη πριν και μετα την καθε κληση της display().Για να εχουμε αμοιβαιο αποκλεισμο ξεκινοντας αρχικοποιω τον σημαφορο με την τιμη 1 κανοντας UP(sem) με την εντολη semop(my_sem, &up, 1); .Επειτα πριν απο καθε κληση της display() κανω DOWN(sem) μεσω της semop(my_sem, &down, 1).Με αυτον τον τροπο αν βρισκομαι στην πρωτη κληση της display() επειδη ο σημαφορος εχει αρχικοποιηθει με την τιμη 1 η αν λαβει την τιμη 1 που σημαινει οτι δεν τρεχει το ιδιο μερος κωδικα αλλη διεργασια τοτε απλα η τιμη γινεται 0 και γινεται κληση της display(),αν ομως η τιμη του σημαφορου ειναι ηδη 0 γιατι εχει γινει DOWN σε διεργασια που τρεχει παραλληλα τοτε μπλοκαρεται η διεργασια μεχρι να τελειωσει αυτη που τρεχει ηδη.Μετα το τελος της display() κανω UP() ωστε αν υπαρχουν μπλοκαρισμενες διεργασιες τις ξεμπλοκαρω αλλιως απλα αυξανω τον σημαφορο.Στο τελος καταστρεφω το σημαφορο πριν το τελος του προγραμματος.

2)Σε αυτο το ερωτημα πρεπει να τυπωνεται πρωτα το ab ακολουθουμενο απο το cd\n .Για να το πετυχω αυτο κανω χρηση 2 σημαφορων .Οι 2 σημαφοροι υλοποιουν συγχρονισμο διεργασιων ως εξης.Αρχικα ο πρωτος σημαφορος αρχικοποιειται με την τιμη 0 και ο δευτερος με την τιμη 1. Στην διεργασια πατερα πριν την κληση της display("ab") μεσω της semop(my_sem, &down2, 1) κανω DOWN(sem) στον 2ο σημαφορο .Αν ειναι η πρωτη επαναληψη ή αν εχει εμφανιστει το "cd\n" απο την διεργασια παιδι (δηλαδη ο 2ος σημαφορος να ειναι 1) τοτε μεταβαινω στην επομενη εντολη που ειναι η display() και εμφανιζεται το "ab" αλλιως αν ο σημαφορος ειναι 0 τοτε περιμενω να εμφανιστει το "cd\n" απο την διεργασια παιδι για να συνεχισω.Επειτα κανω UP(sem) στο 1ο σημαφορο μεσω της semop(my_sem, &up, 1) δηλωνοντας ετσι οτι εχει εμφανιστει το "ab".Στην διεργασια παιδι πριν την κληση της display("cd\n") μεσω της semop(my_sem, &down, 1) κανω DOWN(sem) στον 1ο σημαφορο .Αν εχει εμφανιστει το "ab" απο την διεργασια πατερα (δηλαδη ο 1ος σημαφορος να ειναι 1) τοτε μεταβαινω στην επομενη εντολη που ειναι η display() και εμφανιζεται το "cd\n" αλλιως αν ειμαι στην πρωτη επαναληψη η αν δεν εχει εμφανιστει το "ab" (ο σημαφορος ειναι 0) τοτε περιμενω να εμφανιστει το "ab" απο την διεργασια πατερα για να συνεχισω.Επειτα κανω UP(sem) στο 2ο σημαφορο μεσω της semop(my_sem, &up2, 1) δηλωνοντας ετσι οτι εχει εμφανιστει το "cd\n".

3)Για να μετατραπει το προγραμμα ωστε να χρησιμοποιει νηματα αντι για διεργασιες αρχικα δημιουργηθηκε μια συναρτηση func η οποια υλοποιει το βρογχο for για 10 επαναληψεις και εκτελει την display() για καθε επαναληψη με ορισμα το ορισμα που δεχεται η func.Στην main αρχικα δημιουργω ενα πινακα των 2 threads.Επειρα τα string "Hello world\n" και "Kalimera kosme\n" αποθηκευονται σε 2 μεταβλητες str1 και str2 αντιστοιχα .Επειτα δημιουργω το πρωτο thread ,στην θεση 0 του πινακα των 2 threads, χωρις attributes που εκτελει την συναρτηση func με ορισμα το str1 που περιεχει το "Hello world\n" .Αμεσως μετα δημιουργω το δευτερο thread ,στην θεση 1 του πινακα

των 2 threads, χωρις attributes που εκτελει την συναρτηση func με ορισμα το str2 που περιεχει το "Kalimera kosme\n".Επειτα με την εντολη pthread_join(id1[0], NULL) περιμενω να τελειωσει το thread στη θεση 0 του πινακα που μπορει να εχει σταματησει.Αμεσως μετα ομοιως για το thread στην θεση 1 του πινακα μεσω της pthread_join(id1[1], NULL).Ο αμοιβαιος αποκλεισμος για την αποφυγη ανακατεματος μηνυματων υλοποιειται με τη χρηση mutex στην συναρτηση func ως εξης:Πριν την display() κανω lock το mutex μεσω της pthread_mutex_lock(&mutex).Αν εχει γινει lock απο αλλο thread τοτε περιμενω να γινει ξανα διαθεσημο δηλαδη να γινει unlock απο το αλλο thread.Μετα την display() κανω unlock το mutex αφου τελειωσω με την εκτελεση της κρισημης περιοχης δηλαδη της κλησης του display().Τελος γινεται καταστροφη του mutex πριν το τελος της main.

4) Για να μετατραπει το προγραμμα ωστε να χρησιμοποιει νηματα αντι για διεργασιες αυτη τη φορα δημιουργηθηκαν 2 συναρτησεις func1 και func2 η οποιες υλοποιουν το βρογχο for για 10 επαναληψεις και εκτελουν την display() για καθε επαναληψη με ορισμα το ορισμα str1 η func1 και str2 η func2.Στην main γινεται οτι και στο προηγουμενο ερωτημα με τη διαφορα οτι το str1 περιεχει το “ab” και το str2 το ‘’cd\n” και οτι τα 2 thread δημιουργουνται ετσι ωστε το πρωτο thread να εκτελει την συναρτηση func1 με ορισμα το str1 και το δευτερο την func2 με ορισμα το str2.Για να τυπωνεται το ab ακολουθουμενο παντα απο το cd\n χρησιμοποιουνται condition variables με τον εξης τροπο :Αρχικα η μεταβλητη predicate αρχικοποιειται με την τιμη 1 και η predicate2 με την τιμη 0.

Στην func1 πριν την for γινεται lock το mutex μεσω της pthread_mutex_lock(&mutex).Μεσα στον βρογχο for πριν την display() δημιουργω ενα βρογχο while (predicate==0) pthread_cond_wait(&cond_var,&mutex) .Στην πρωτη επαναληψη το predicate εχει την τιμη 1 αρα δεν μπαινει στο while και εμφανιζεται μεσω της display() το “ab”.Στις υπολοιπες επαναληψεις αν το predicate ειναι ισο με 0 που σημαινει οτι δεν εχει εμφανιστει το “cd\n” τοτε περιμενω μεχρι να εμφανιστει για να συνεχισω (δηλαδη μεχρι να γινει το predicate ισο με 1)μεσω της pthread_cond_wait(&cond_var,&mutex) μενοντας μεσα στο βρογχο while. Μετα την display() δηλαδη αφου τελειωσω την εμφανιση του “ab” θετω το predicate ισο με 0 ωστε στην επομενη επαναληψη να περιμενει την εμφανιση του “cd\n” απο το δευτερο thread και θετω το predicate2 ισο με 1 ωστε να βγει απο το βρογχο while το δευτερο thread και να εκτελεστει η display() που εμφανιζει το “cd\n”.Επειτα με την pthread_cond_signal(&cond_var) ξεμπλοκαρω τα νηματα που εχουν μπλοκαριστει απο μια condition variable.Μετα το τελος της for κανω unblock το mutex με την pthread_mutex_unlock(&mutex).

Στην func2 πριν την for γινεται lock το mutex μεσω της pthread_mutex_lock(&mutex).Μεσα στον βρογχο for πριν την display() δημιουργω ενα βρογχο while (predicate2==0) pthread_cond_wait(&cond_var,&mutex) .Στην πρωτη επαναληψη το predicate2 εχει την τιμη 0 αρα περιμενω μεχρι να εκτελεστει πρωτα το πρωτο thread που εμφανιζει το ‘’ab” μεσω της display().Στις υπολοιπες επαναληψεις αν το predicate2 ειναι ισο με 0 που σημαινει οτι δεν εχει εμφανιστει το “ab” τοτε περιμενω μεχρι να εμφανιστει για να συνεχισω (δηλαδη μεχρι να γινει το predicate2 ισο με 1) μεσω της pthread_cond_wait(&cond_var,&mutex) μενοντας μεσα στο βρογχο while.Μετα την display() δηλαδη αφου τελειωσω την εμφανιση του “cd\n” θετω το predicate ισο με 1 ωστε να βγει απο το βρογχο while το πρωτο thread και να εκτελεστει η display() που εμφανιζει το “ab” και θετω το predicate2 ισο με 0 ωστε στην επομενη επαναληψη να περιμενει την εμφανιση του “ab” απο το πρωτο thread.Επειτα με την pthread_cond_signal(&cond_var) ξεμπλοκαρω τα νηματα που εχουν μπλοκαριστει απο μια condition variable. Μετα το τελος της for κανω unblock το mutex με την pthread_mutex_unlock(&mutex).