Αρχικά, αφού ορίσαμε όλες τις μετάβλητες μας φτιάξαμε τη συνάρτηση service η οποία εκτελεί όλη τη διαδικασία πελάτη-τηλεφωνητή. Πιο συγκεκριμένα, μέσω της συνάρτησης

clock_gettime(CLOCK_REALTIME ,&start) υπολογίζει την χρονική στιγμή που ένας πελάτης εμφανίστηκε και αμέσως αφαιρεί αυτή τη χρονική στιγμή από τους μέσους χρόνους εξυπηρέτησης και αναμονής.Στη συνέχεια, με χρήση ενός mutex (rc= pthread_mutex_lock(&mutex);) κλειδώνεται η κρίσιμη περιοχή στην οποία ένας πελάτης αναμένει για εξυπηρέτηση από κάποιον διαθέσιμο τηλεφωνητή και για όσο χρονικό διάστημα η μεταβλητη tel_count είναι ίση με 0 τότε το συγκεκριμένο Thread(πελάτης) περιμένει μέχρι η μεταβλητη να αυξηθεί που σημαίνει ότι ελευθερώθηκε κάποιος τηλεφωνητής και μπορεί να τον εξυπηρετήσει και παλι ξαναμειώνεται ώστε να φανεί ότι ο πελάτης βρηκε τηλεφωνητή. Μέσω πάλι της συναρττησης clock_gettime(CLOCK_REALTIME,&stop) υπολογίζεται η χρονική στιγμη κατά την οποία ο πελατης βρήκε διαθέσιμο τηλεφωνητή και στον μέσο χρόνο αναμόνης προστίθεται αυτή η χρονική στιγμη και τότε μέσω της sleep(rand_r(&arguments->seed) %(T_SEATHIGH - T_SEATLOW +1) + (T_SEATHIGH - T_SEATLOW) ); Επιστρέφεται ο τυχαίος αριθμός που αντιστοιχεί στα δευτερόλεπτα που χρειάστηκε ο τηλεφωνητης για να ψάξει για τις διαθέσιμες θέσεις που μπορεό να πουλήσει στον πελάτη. Έπειτα, κλειδώνεται η κρίσιμη περιοχη στην οποια γινεται ο ΄σλεγχος για αν υπάρχουν διαθέσιμα καθίσματα και εκτελείται και εκτελείται λίγο πρίν τον λεγχο η pthread_cond_wait(&cond, &mutex); Ώστε να μην υπάρξει κάποιο πρόβλημα από μη ενημέρωση του πλάνου των θέσεων. Αφού ελέγξει ότι υπάρχουν διαθέσιμες θέσεις τότε συνεχίζει στην εκτέλεση της διαδιακσιας πώλησης των θέσεων.

Γίνεται ένα ακόμη κλείδωμα π΄ριν γίνει η τοποθέτηση του id του thread στο πλάνο θέσεων λόγω της πώλησης τους σε αυτόν (ουσιαστικά ο πίνακας με το πλάνο των θέσεων εχει ως στοιχεία τα id των thread στα οποία εχουν πουληθέι οι αντίστοιχες θέσεις ).

Μετά γίνεται ένα ακομη κλείδωμα κατά την εκ΄τελεση της συναλλαγης και άλλο ένα στην πείπτωση που υπάρχει πρόβλημα με τη συναλλαγή και η κρλατηση ακυρώνεται.

Στη συνέχεια γίνεται κλείδωμα π΄ριν υπολογιστεί η χρονικη στιγμη της ολοκλήρωσης της εξυπηρέτησης μέσω της clock_gettime(CLOCK_REALTIME,&stop) .

Τέλος γίνεται ένα ακόμη κλείδωμα για την αύξηση της μεταβλητης τηλεφωνητη όταν ένα thread έχει εξυπηρετηθει και εχει γινει η εξοδος του.

Ακολουθει η main στην οποια γινεται η δημιουργια των thread και μέσω της service γίνεται όλη η απιτουμενη διαδικασια.

Γενικα, στο πρόγραμμα μας εχουμε βαλει 4 διαφορετικα mutexes ώστε όταν ένα Thread μπαίνει σε μια κρισιμη περιοχη όλα τα υπόλοιπα να μπλοκαρονται ενώ τα υπόλοιπα τμήματα του προγράμματος να συνεχίζουν να εκτελούνται από διαφορετικά νήματα τη φορά.