**Ουρά queue**

**Συνάρτηση server()**

Η συγκεκριμένη συνάρτηση αρχικοποιεί το σύστημα για την λήψη μηνυμάτων.

Ανοίγει καινούργιο socket και το αρχικοποιεί περιμένοντας μηνύματα από άλλα συστήματα στο ίδιο δίκτυο.

Μέσω της συνάρτησης listen() επιτρέπουμε στο σύστημα να συνδεθεί με εως 3 συσκευές ταυτόχρονα.

Έπειτα η συνάρτηση δέχεται το μήνυμα μέσω της συνάρτησης accept() και το διαβάζει με τη συνάρτηση read() και στην πορεία το εκτυπώνει.

ΜΗΠΩΣ ΠΡΕΠΕΙ ΑΝΤΙ ΓΙΑ READ ΝΑ ΕΙΝΑΙ RECV???

ΙΣΩΣ ΘΕΛΕΙ ΩΣ ΟΡΙΣΜΑ ΜΟΝΟ ΤΟ ΑΝΤΡΕΣΣΕΙΣ ΚΑΙ ΟΧΙ ΤΟ ΤΟΥΓΚΕΔΕΡ

**Συνάρτηση timer()**

Η συνάρτηση αυτή εισάγει μια καθυστέρηση 10 δευτερολέπτων ανάμεσα στις αναζητήσεις των mac διευθύνσεων. Ουσιαστικά ανά 10 δευτερόλεπτα καλεί μια συνάρτηση για την εύρεση των διευθύνσεων.

Η συναρτηση αυτή έχει επίσης τρεις μετρητές counter, counter2 και counter3 αντίστοιχα.

Ο πρώτος αυξάνεται και όταν φτάσει να μετρήσει 22 λεπτά στέλνει σήμα μέσω του wait\_timer signal στην συνάρτηση find\_close\_contacts που περιμένει.

Ο δεύτερος αντίστοιχα μετράει 14 μέρες ώστε να στείλει σήμα μέσω του delete\_close\_wait signal στην συνάρτηση delete\_close\_contacts.

Τελικά ο τρίτος μετράει 4 ώρες ώστε να στείλει σήμα μέσω του test\_wait signal στη συνάρτηση test.

**Συνάρτηση find\_mac()**

Η συνάρτηση αυτή καταρχάς επιλέγει με τυχαίο βαθμό αν έχει βρεθεί κάποια mac διεύθυνση σε απόσταση αρκετά κοντινή για να την ανακαλύψουμε και αν αποφασίσει ότι υπάρχει, επιλέγει μια τυχαία τιμή από τον πίνακα με τις Mac διευθύνσεις που έχει αρχικοποιηθεί στην αρχή του προγράμματος. Έπειτα αποθηκεύει στην ουρά addresses τον χρόνο της επαφής, την διεύθυνση mac και τέλος την θέση της στον αρχικό πίνακα Macs.

**Ιδανικά: το πρόγραμμα ελέγχει με εντολή arp-a και αποθηκεύει τις mac διευθύνσεις των συσκευών που βρίσκονται στο ίδιο δίκτυο.**

**Συνάρτηση find\_close\_contacts()**

Η συνάρτηση αυτή αρχικά περιμένει ένα σήμα από το wait\_timer signal. Δηλαδή περιμένει την συνάρτηση timer να την ενημερώσει πότε θα είναι η ώρα της να τρέξει. Για την έυρεση κοντινών επαφών η εκφώνηση της άσκησης ζητάει να έχουν περάσει 4-20 λεπτά με επαναλαμβανόμενη επαφή για κάθε mac διεύθυνση.

Επομένως η συνάρτηση καταρχάς ελέγχει ότι η ουρά με τις διευθύνσεις δεν είναι άδεια. Έπειτα ελέγχει μήπως το πρώτο στοιχείο της βρίσκεται στην ουρά για παραπάνω από 41 λεπτά και το διαγράφει (η συνάρτηση θα είχε χρόνο να το βρει ακόμα και στις οριακές του τιμές αν ξαναυπήρχε). Έπειτα ξεκινάει να ψάχνει μεσα στην ουρά για mac διευθύνσεις που επαναλαμβάνονται. Η έρευνα γίνεται με τον συγκρίνουμε το πρώτο στοιχείο με το τελευταίο και έπειτα το δεύτερο με το τελευταίο και ούτε καθεξής. Αν το πρώτο στοιχείο στην ουρά έχει λιγότερο από 4 λεπτά που βρίσκεται στην ουρά κατευθείαν η έρευνα σταματάει. Η αναζήτηση γίνεται ως εξής:

Σε περίπτωση που βρεθεί η ίδια mac στην ουρά δύο φορές το πρόγραμμα συγκρίνει το timestamp τους, δηλαδή το πόση ώρα έχει περάσει από την πρώτη φορά που έχει σώσει την mac αυτή με την τελευταία. Σε περίπτωση που αυτό το timestamp είναι μεγαλύτερο των 20 λεπτών που ζητάει η εκφώνηση το πρώτο στοιχείο της επαφής αυτής διαγράφεται από την ουρά. Σε περίπτωση που το timestamp είναι μεγαλύτερο από 4 λεπτά και μικρότερο από 20 το πρώτο στοιχείο της επαφής αυτής προστίθεται στην δεύτερη ουρά με την συνάρτηση save\_close\_contact. Επίσης αποθηκεύεται και η mac της διεύθυνση στην μεταβλητή history έτσι ώστε να μην υπάρξουν διπλοεγγραφές των επαφών στο συγκεκριμένο κομμάτι εύρεσης τους. Έπειτα διαγράφουμε το πρώτο στοιχείο της επαφής από την αρχική ουρά καθώς δεν θα το χρησιμοποιούσουμε περαιτέρω. Στην τελευταία περίπτωση που το timestamp είναι μικρότερο από 4 λεπτά αποχωρούμε από την διαδικασία εύρεσης κοντινών επαφών και περιμένουμε ξανά να μας καλέσει η συνάρτηση timer για να ξαναρχίσουμε.

Σε περίπτωση που δεν βρεθεί η ίδια mac στην ουρά μετακινούμαστε πιο αριστερά από την ουρά μέχρι είτε να βρεθεί ίδια mac είτε να εξαντλήσουμε τους πιθανούς συνδιασμούς της ουράς.

**Συνάρτηση save\_close\_contacts()**

Η συνάρτηση αυτή αποθηκεύει τις επαφές που θεωρήθηκαν κοντινές σε μια δεύτερη ουρά την close\_contacts.

**Συνάρτηση delete\_close\_contacts()**

Η συνάρτηση αυτή διαγράφει τις κοντινές επαφές οι οποίες βρίσκονται στην ουρά για παραπάνω από 14 μέρες. Καταρχάς η συνάρτηση περιμένει σήμα από την συνάρτηση timer() ππου σημαίνει ότι έχουν περάσει 14 μέρες από τη στιγμή που επαναλήφθηκε η συγκεκριμένη διαδικασία. Η συνάρτηση ελέγχει κάθε μια από τις κοντινές επαφές ξεκινώντας από την πρώτη στην ουρά και σε περίπτωση που το timestamp τους είναι μεγαλύτερο από 14 μέρες τις διαγράφει. Σε περίπτωση που φτάσουμε σε επαφή που το timestamp της είναι μικρότερη από 14 μέρες το πρόγραμμα τερματίζει μέχρι να το ξανακαλέσουμε, αφού δεν υπάρχει περίπτωση να υπάρχει μετέπειτα στην ουρά επαφή με μεγαλύτερο timestamp.

**Συνάρτηση test()**

Η συνάρτηση αυτή δημιουργήθηκε για να δίνει αποτελέσματα για την εξέταση covid του χρήστη. Καταρχάς περιμένει 4 ώρες μέχρι δηλαδή να του έρθει σήμα από τη συνάρτηση timer() και έπειτα διενεργεί το τεστ. Αν τα αποτελέσματα του τεστ είναι αρνητικά τερματίζει μέχρι να ξανακαλεστεί. Αντίθετα αν τα αποτελέσματα του τεστ είναι θετικά πρέπει να ενημερώσει τις κοντινές επαφές οπότε καλεί την συνάρτηση save\_server().

**Συνάρτηση save\_server()**

Η συνάρτηση αυτή καλείται όταν είτε το δικό μας τεστ covid έχει βγει θετικό, είτε όταν έχουμε έρθει σε επαφή με επιβεβαιωμένο κρούσμα. Η συνάρτηση δημιουργεί ένα thread που καλεί την συνάρτηση client η οποία θα επικοινωνήσει με τις κοντινές επαφές για να τις ενημερώσει για την κατάσταση. Έπειτα θα σώσει τις επαφές σε ένα binary αρχείο. Ο λόγος που δεν διαγράφει κιόλας τις επαφές είναι γιατί μπορεί να υπάρξει conflict με την επικοινωνία.

χρησιμοποιούμε signals καθώς δεν θέλουμε να τρώμε δυνάμεις από τη cpu

***MAIN***

* *αρχικοποίηση addresses, closeContacts μάλλον πρέπει να είναι και τα δυο τύπου queue*
* *αρχικοποίηση threads και mutexes*
* *καλεί συναρτήσεις μέσω threads για να τρέχουν παράλληλα*