

Sockets with C

Περιεχόμενα

Εκφώνηση	2
Αρχεία εργασίας και περιγραφή	3
Βήματα εκτέλεσης	4
Notes && Links.....	5

Εκφώνηση

Υποθέτουμε πως έχουμε στη διάθεσή μας 3 εξυπηρετητές και 20 πελάτες. Οι 2 εξυπηρετητές έχουν στη διάθεσή τους δεδομένα με τη μορφή διανυσμάτων (π.χ., 1000 διανύσματα). Ο τελευταίος εξυπηρετητής αναλαμβάνει να λειτουργήσει ως ενδιάμεσος μεταξύ των πελατών και των δύο εξυπηρετητών. Ανά τακτά διαστήματα οι δύο εξυπηρετητές ενημερώνουν τον τρίτο για το μέσο όρο και τη διακύμανση των δεδομένων (της κάθε διάστασης). Ο καθένας από αυτούς στέλνει δύο διανύσματα, ένα με τους μέσους όρους και ένα με τη διακύμανση. Για παράδειγμα θεωρήστε τα ακόλουθα διανύσματα: , . Συνεπώς, ο 'ενδιάμεσος' εξυπηρετητής έχει στη διάθεσή του τέσσερα διανύσματα. Οι πελάτες παράγουν τιμές-διανύσματα τα οποία θέλουν να τοποθετήσουν στο σωστό εξυπηρετητή. Για να το κάνουν αυτό θα πρέπει να συμβουλευτούν τον ενδιάμεσο εξυπηρετητή. Συνεπώς, κάθε φορά που παράγουν ένα διάνυσμα, στέλνουν ένα μήνυμα με τα δεδομένα τους στον ενδιάμεσο και αυτός τους αποκρίνεται με το ποιος από τους δύο άλλους εξυπηρετητές είναι ο κατάλληλος για να αποθηκεύσει τα δεδομένα. Η απόφαση λαμβάνεται με τη βοήθεια της ομοιότητας των δεδομένων με τα διανύσματα των μέσων όρων και των διακυμάνσεων. Ο ενδιάμεσος θα εφαρμόσει μια Ευκλείδεια απόσταση με τους μέσους όρους και θα βρει ποια απόσταση είναι η μικρότερη. Στη συνέχεια, θα υπολογίσει το μέσο όρο όλων των διακυμάνσεων. Ο καταλληλότερος προς αποθήκευση εξυπηρετητής είναι αυτός με τη μικρότερη απόσταση εκτός και αν η μέση τιμή των διακυμάνσεων υπερβαίνει ένα προκαθορισμένο όριο. Ο ενδιάμεσος αφού βρει τον καταλληλότερο εξυπηρετητή, θα αποστείλει κατάλληλο μήνυμα προς τον πελάτη και αυτός στη συνέχεια θα προωθήσει τα δεδομένα προς τον επιλεγμένο εξυπηρετητή. Ο εξυπηρετητής μετά την αποθήκευση στέλνει μήνυμα αποδοχής προς τον πελάτη ώστε να κλείσει η επικοινωνία. Να προσομοιώσετε το παραπάνω σενάριο με τη βοήθεια των sockets καθώς και ότι άλλο κρίνετε ότι είναι χρήσιμο για την υλοποίηση.

Σημείωση: Η υλοποίηση της άσκησης δεν έχει πραγματοποιηθεί στο 100%. π.χ. η διαδικασία για την διακύμανση δεν έχει υλοποιηθεί σωστά.

Αρχεία εργασίας και περιγραφή

Τα κύρια αρχεία τις εργασίας είναι τέσσερα:

- Main Server
- Server A
- Server B
- Client

Κατά την εκτέλεση αυτών των αρχείων θα δημιουργηθούν κάποια .txt αρχεία. Όλα τα c αρχεία έχουν κάποιο κοινό κώδικα για τα sockets οπότε μην τρομάξεις από το μέγεθος του κώδικα. Έχω σε comments κάποιες printf εντολές έτσι ώστε να τυπώσεις ανά διαστήματα δεδομένα για να καταλάβεις καλύτερα τι γίνεται.

Ο κώδικας αυτό που κάνει είναι :

Ο πελάτης δημιουργεί array με τυχαίο μέγεθος και περιέχει τυχαίους αριθμούς. Αυτό το array είναι το διάνυσμα μας το οποίο στέλνεται στον main server. Όταν πάρει απάντηση από τον main server μέσω μια function συνδέεται στον κατάλληλο server στέλνοντας του το ίδιο array.

Ο main server δέχεται το array υπολογίζει την ευκλείδεια απόσταση και το μέσο όρο των διακυμάνσεων(Με τον μέσο όρο των διακυμάνσεων δεν κάνω κάποιο υπολογισμό). Αφού βρεθεί η ευκλείδεια απόσταση ανάλογα με τα αποτελέσματα στέλνεται μήνυμα στον πελάτη .

Ο serverA-B δέχεται ένα array και το καταχωρεί σε ένα αρχείο (μαζί με τα άλλα διανύσματα, κάθε γραμμή και ένα διάνυσμα) αφού το καταχωρίσει υπολογίζει το μέσο όρο και την διακύμανση γραμμή –γραμμή, έπειτα αποθηκεύει τα αποτελέσματα σε αρχεία.

Βήματα εκτέλεσης

Τα βήματα εκτέλεσης των αρχείων είναι τα εξής:

1. Πρώτα κάνουμε compile τα αρχεία μας με τις εντολές
gcc {όνομα αρχείου}.c -o {όνομα μεταγλωττισμένου αρχείου} -lm
παράδειγμα :
το -lm χρειάζεται μόνο όταν κάνεις compile τους servers.
2. Επόμενο βήμα είναι να τρέξεις τα αρχεία. Πρώτα τους Servers και μετά τον client
Με την εντολή ./{όνομα μεταγλωττισμένου αρχείου}
Παράδειγμα :

```
./mainserver
```

3. Για να πάνε όλα καλά πρέπει να υπάρχουν τα txt AvgFromServerA και AvgFromServerB. Με κάποιο υποτιθέμενο μέσο όρο και τα txt FileA και FileB όπου αποθηκεύονται τα διανύσματα.
4. Οπότε βάζεις τους servers να τρέχουν και απλά εκτελείς τον client όσες φορές θέλεις.

Notes & Links

- [Online calculator](#) πολύ βοηθητικό.
- [Διακύμανση](#) Θεωρία[fluctuation].

Για ένα πεπερασμένο σύνολο αριθμών, η συνήθης απόκλιση βρίσκεται λαμβάνοντας την τετραγωνική ρίζα του μέσου όρου των τετραγώνων των αποκλίσεων των τιμών από τη μέση τιμή τους. Για παράδειγμα, οι βαθμοί από μια τάξη οχτώ μαθητών (δηλαδή, ένας πληθυσμός) είναι οι ακόλουθες οκτώ τιμές:

2, 4, 4, 4, 5, 5, 7, 9.

Αυτά τα οκτώ στοιχεία έχουν την μέση τιμή (μέσο όρο) 5:

$$\frac{2 + 4 + 4 + 4 + 5 + 5 + 7 + 9}{8} = 5.$$

Πρώτα, υπολογίζονται τα τετράγωνα των αποκλίσεων του κάθε στοιχείου από τη μέση τιμή:

$$\begin{aligned}(2 - 5)^2 &= (-3)^2 = 9 & (5 - 5)^2 &= 0^2 = 0 \\(4 - 5)^2 &= (-1)^2 = 1 & (5 - 5)^2 &= 0^2 = 0 \\(4 - 5)^2 &= (-1)^2 = 1 & (7 - 5)^2 &= 2^2 = 4 \\(4 - 5)^2 &= (-1)^2 = 1 & (9 - 5)^2 &= 4^2 = 16.\end{aligned}$$

Η διακύμανση είναι ο μέσος των τιμών αυτών:

$$\frac{9 + 1 + 1 + 1 + 0 + 0 + 4 + 16}{8} = 4.$$

- [Ευκλείδεια απόσταση](#) για μια διάσταση.
- Ευκλείδεια απόσταση [παράδειγμα](#).

Δύο διαστάσεις [[Επεξεργασία](#) | [επεξεργασία κώδικα](#)]

Για δύο δισδιάστατα σημεία στο επίπεδο, $P = (p_x, p_y)$ και $Q = (q_x, q_y)$, η Ευκλείδεια απόσταση είναι:

$$\sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}$$

- [Pow](#), [Sqrt](#), [Fabs](#), [htonl](#).