

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Άσκηση 3 “Προγραμματισμός Συστήματος” 2007-08 (Προθεσμία 09/01 – 18:00)

Απομακρυσμένη Εκτέλεση Εντολών.

Στόχος: Εξουκείωση με χρήση υποδοχών (sockets) και σωλήνων (pipes).

ΠΡΟΣΟΧΗ: Η ΑΣΚΗΣΗ ΜΠΟΡΕΙ ΝΑ ΥΛΟΠΟΙΗΘΕΙ ΕΙΤΕ ΑΤΟΜΙΚΑ ΕΙΤΕ ΣΕ ΟΜΑΔΕΣ ΤΩΝ 2 ΑΤΟΜΩΝ. ΣΕ ΑΥΤΗ ΤΗΝ ΑΣΚΗΣΗ ΠΑΡΑΤΗΡΗΣΤΕ ΠΑΡΑΚΑΤΩ ΟΤΙ ΟΙ ΑΠΑΙΤΗΣΕΙΣ ΕΙΝΑΙ ΔΙΑΦΟΡΕΤΙΚΕΣ (ΕΛΑΦΡΩΣ ΜΙΚΡΟΤΕΡΕΣ) ΓΙΑ ΑΥΤΟΥΣ ΠΟΥ ΘΑ ΤΗΝ ΥΛΟΠΟΙΗΣΟΥΝ ΜΟΝΟΙ ΤΟΥΣ. ΠΡΟΣΟΧΗ ΣΤΙΣ ΑΠΑΙΤΗΣΕΙΣ ΤΗΣ ΑΣΚΗΣΗΣ. ΔΕ ΘΑ ΓΙΝΟΥΝ ΔΕΚΤΕΣ ΕΡΓΑΣΙΕΣ ΜΕ ΔΙΑΦΟΡΕΤΙΚΗ ΥΛΟΠΟΙΗΣΗ ΤΩΝ ΑΠΑΙΤΗΣΕΩΝ. ΟΣΟΙ ΔΕΝ ΠΡΟΣΕΛΘΟΥΝ ΝΑ ΕΞΕΤΑΣΤΟΥΝ ΔΕ ΘΑ ΛΑΒΟΥΝ ΒΑΘΜΟ, ΑΚΟΜΑ ΚΑΙ ΑΝ ΕΧΟΥΝ ΠΑΡΑΔΩΣΕΙ ΤΗΝ ΑΣΚΗΣΗ.

Περιγραφή Λειτουργίας Εξυπηρετητή

Το πρώτο πρόγραμμα, με το όνομα remoteServer θα εκτελεί ένα περιορισμένο υποσύνολο από εντολές του Unix που θα του στέλνουν οι πελάτες. Οι εντολές αυτές θα είναι οι εντολές ls, cat, cut, grep και tr. Όλες αυτές οι εντολές μπορεί να περιέχουν παραμέτρους (πχ: ls -alR /usr/) και σωληνώσεις, αρκεί η πρώτη εντολή κάθε σωληνώσεως να είναι ανάμεσα σε αυτές τις 5 εντολές (εννοείται ότι μπορούν να υπάρχουν κενά πριν την κάθε εντολή). **Εντολές με πάνω από 256 χαρακτήρες ή μη υλοποιημένες εντολές διαβάζονται αλλά αγνοούνται.**

Όποιος μπορέσει να υλοποιήσει και κάποιο στοιχειώδη έλεγχο για την ορθότητα των εντολών όπως περιγράφεται σε αυτή την παράγραφο, θα λάβει κάποιο μικρό **extra bonus**. Σε περίπτωση σωληνώσεων με κάποια διαφορετική εντολή από τις παραπάνω 5, αγνοείται το κομμάτι από τη μη υλοποιημένη εντολή και πέρα. Σε ακολουθίες εντολών με ερωτηματικά εκτελείται μόνο η πρώτη από τις εντολές, αν αυτή είναι ανάμεσα στις 5 εντολές. Αλλιώς επιστρέφεται μήνυμα σφάλματος στον πελάτη. (Επαναλαμβάνεται ότι αυτοί οι έλεγχοι αφορούν extra bonus). Για παράδειγμα:

Εντολή	ΣΧΟΛΙΑ	Επιστρέφεται στον πελάτη το αποτέλεσμα...
ls -alR /etc/	Σωστή εντολή	ls -alR /etc/
cut -f1,3 koko tr ‘:’ ‘ ’	Σωστή εντολή	cut -f1,3 koko tr ‘:’ ‘ ’
cut -f1,3 koko; rm -i kokoDir	Σύνθετη εντολή	cut -f1,3 koko
rm -i kokoDir; cut -f1,3 koko	Μη υποστηριζόμενη εντολή rm	Μήνυμα σφάλματος
ls -alR /etc/ rm -i	Μη υποστηριζόμενη εντολή rm	ls -alR /etc/

Δομή Εξυπηρετητή

Το πρόγραμμα remoteServer παίρνει 3 παραμέτρους. Η πρώτη παράμετρος (PORT) είναι ο αριθμός της θύρας όπου ο εξυπηρετητής θα ακούει για TCP συνδέσεις από πελάτες. Η δεύτερη παράμετρος (NUMCHILDREN >= 1) θα δηλώνει πόσα παιδιά θα γεννήσει η κύρια διεργασία για την εξυπηρέτηση των πελατών. Η τρίτη παράμετρος (probFail) δηλώνει με ποια πιθανότητα ένα τηλεγραφικό μήνυμα (datagram) μπορεί να χαθεί.

Όταν ξεκινάει το πρόγραμμα, ο εξυπηρετητής εκτελεί όλες τις διαδικασίες για να δέχεται TCP συνδέσεις και γεννάει NUMCHILDREN διεργασίες παιδιά. **Όλες οι διεργασίες του εξυπηρετητή (πατέρας και παιδιά) να αγνοούν το σήμα SIGPIPE.** Όλες οι αιτήσεις για

σύνδεση εξυπηρετούνται ΜΟΝΟ από τη διεργασία πατέρα. Όλες οι εντολές προς εκτέλεση από τους πελάτες διαβάζονται από τη διεργασία πατέρα (θυμηθείτε ότι αυτές με πάνω από 256 χαρακτήρες διαβάζονται αλλά αγνοούνται). Αυτές οι εντολές μοιράζονται μετά από τον πατέρα στα παιδιά προς εκτέλεση.

Το μοίρασμα αυτό θα πρέπει να γίνεται με τρόπο ώστε να λαμβάνεται υπ όψιν το φορτίο κάθε παιδιού. Πχ, αν έχουμε 2 παιδιά, μια διαδικασία που αναθέτει τα μονά ερωτήματα στο πρώτο παιδί και τα ζυγά στο δεύτερο δεν λαμβάνει υπ όψιν πόσο χρόνο απαιτεί η εκτέλεση αυτών των ερωτημάτων. Πιο σωστή (και ζητούμενη από την άσκηση για τις ομαδικές εργασίες) είναι η καταγραφή των ερωτημάτων σε ένα σωλήνα (pipe) προσπελάσιμο από όλα τα παιδιά. Το πρώτο διαθέσιμο παιδί θα εκτελεί πάντα την πρώτη εντολή που θα βρίσκει στο σωλήνα.

Σε αυτή τη διαδικασία είναι απαραίτητο να γνωρίζετε πώς συμπεριφέρονται οι συναρτήσεις **read** και **write** σε σωλήνες. Θα χρειαστείτε σίγουρα να συγχρονίσετε την ανάγνωση των μηνυμάτων. Αυτός ο συγχρονισμός να γίνει ΜΟΝΟ με χρήση σωλήνων και όχι με άλλους τρόπους που δεν έχουμε αναφέρει στο μάθημα μέχρι την ανακοίνωση της άσκησης αυτής (κοινή μνήμη, σηματοφορείς, υποδοχές στο πεδίο UNIX κτλ). Για να σας βοηθήσω σε αυτό το θέμα, σκεφτείτε ότι μπορεί να υπάρχει σε ένα σωλήνα μία σημαία (flag). Όταν ένα παιδί θέλει να διαβάσει ένα μήνυμα, πρώτα αποκτάει τη σημαία, διαβάζει ΟΛΟ το μήνυμα και μετά τοποθετεί πίσω τη σημαία. Πόσο μεγάλη σε μέγεθος θα πρέπει να είναι αυτή η σημαία, ώστε να τη διαβάζει ΠΑΝΤΑ ατομικά μία read εντολή? Πώς μπορείτε να καταλάβετε ότι κάποια διαδικασία δε θα διαβάσει κατά λάθος αντί για τη σημαία κάποιο κομμάτι ενός μηνύματος; Πόσοι σωλήνες απαιτούνται να υπάρχουν;

Αφού κάποιο παιδί έχει διαβάσει μία εντολή, την εκτελεί και επιστρέφει το αποτέλεσμα στον πελάτη που ρώτησε την εντολή με χρήση τηλεγραφικών υποδοχών (datagram sockets). Αν η απάντηση είναι μεγάλη (>512 bytes), τότε το παιδί στέλνει την απάντηση σε πολλά πακέτα μεγέθους το πολύ 512 bytes το καθένα. Κάθε ένα από αυτά τα πακέτα αποθηκεύεται σε ένα buffer μνήμης (πχ, πίνακα) αλλά δεν στέλνεται στον πελάτη με πιθανότητα probFail (τρίτη παράμετρος του προγράμματος) – αλλιώς στέλνεται στον πελάτη κανονικά.

Πέραν από τις 5 UNIX εντολές, αν η διεργασία παιδί λάβει μία εντολή **end**, τότε ειδοποιεί τον πατέρα του με ένα σήμα (signal), πχ με την εντολή kill, κλείνει ΟΛΟΥΣ τους περιγραφητές του, αποδεσμεύει όλους τους σημαντικούς πόρους της (γιατί τονίζεται αυτό εδώ?) και τερματίζει τη λειτουργία του. Αν η διεργασία παιδί λάβει την εντολή **timeToStop**, τότε ειδοποιεί με κατάλληλο σήμα τον πατέρα του. Ο πατέρας μετά φροντίζει να τερματίσει τη λειτουργία ΟΛΩΝ των παιδιών (αφού αυτά κλείσουν όλους τους περιγραφητές τους) και στη συνέχεια τερματίζει και ο ίδιος τη λειτουργία του, αφού όμως έχουν τερματίσει ΟΛΑ τα παιδιά του. Οποιαδήποτε διεργασία (πατέρας ή παιδί) τερματίζει τη λειτουργία της τυπώνει κατάλληλο μήνυμα που περιλαμβάνει το αναγνωριστικό της (process id) στον περιγραφητή 2 (stderr).

Δομή Πελάτη

Το πρόγραμμα remoteClient παίρνει 3 παραμέτρους. Η πρώτη παράμετρος serverName είναι το όνομα του μηχανήματος στο οποίο τρέχει ο εξυπηρετητής. Η δεύτερη παράμετρος serverPORT δηλώνει τον αριθμό της θύρας όπου ο εξυπηρετητής ακούει για TCP συνδέσεις από πελάτες. Η τρίτη παράμετρος receivePort δηλώνει τη θύρα όπου ο πελάτης θα αναμένει μηνύματα μέσω τηλεγραφικής υποδοχής (datagram socket) ως απαντήσεις στις εντολές που στέλνει. Όλες οι απαντήσεις θα πρέπει να τυπώνονται στην οθόνη, συνοδευόμενα από το αναγνωριστικό της διεργασίας παιδιού (pid) στον εξυπηρετητή που εκτέλεσε την εντολή (στη μορφή εντολή:pid:Αποτέλεσμα εντολής). Όλα τα ερωτήματα (εντολές) στέλνονται στον εξυπηρετητή με βάση την TCP σύνδεση (και οι απαντήσεις λαμβάνονται με τηλεγραφική υποδοχή). Συνεπώς

διαφορετικές διεργασίες-παιδιά του εξυπηρετητή μπορούν να απαντήσουν σε διαφορετικές εντολές του ίδιου πελάτη.

Πράγματα προς σκέψη

- Σκεφτείτε τι πληροφορία θα πρέπει να περνάει η διεργασία πατέρα στα παιδιά για κάθε καινούργια εντολή που λαμβάνει.
- Σκεφτείτε πώς θα διαβάζουν συγχρονισμένα τα παιδιά τα μηνύματα του πατέρα τους. Πώς ξέρουν τα παιδιά πόσο μεγάλο είναι το κάθε μήνυμα;
- Σκεφτείτε πώς θα στέλνουν τα παιδιά απαντήσεις στους πελάτες. Πώς ξέρει ο πελάτης ότι τελείωσε η απάντηση που περίμενε και δεν ακολουθεί και νέο μήνυμα;
- Πώς μπορεί να ξέρει η διεργασία πατέρα του εξυπηρετητή ποιοι πελάτες έχουν στείλει ερωτήματα? Κοιτάξτε ΚΑΛΑ τη συνάρτηση select (man select). Χρησιμεύει για να δείτε ποιοι περιγραφητές έχουν διαθέσιμα δεδομένα χωρίς να μπλοκάρει αν κάπου δεν υπάρχουν δεδομένα. Ουσιαστικά η διαδικασία χρήσης της είναι:
 - ο Δημιουργείτε ένα fd_set στο οποίο θέτεται ένα-ένα τα bits που αντιστοιχούν στον περιγραφητή κάθε ήδη αποδεκτής σύνδεσης και στον περιγραφητή όπου αναμένει για συνδέσεις ο εξυπηρετητής (συναρτήσεις FD_ZERO αρχικά, και μετά διαδοχικά FD_SET).
 - ο Κλήση της select με κατάλληλη readfds παράμετρο
 - ο Αν μετά τη select ο περιγραφητής όπου ο εξυπηρετητής αναμένει συνδέσεις ικανοποιεί την FD_ISSET, τότε υπάρχει νέα αίτηση για σύνδεση.
 - ο Αν μετά τη select ο περιγραφητής κάποιας υπάρχουσας σύνδεσης ικανοποιεί την FD_ISSET, τότε υπάρχουν δεδομένα από αυτόν τον πελάτη.

Βαθμολογία

- Η βαθμολογία θα εξαρτηθεί από το πόσο μεγάλο κομμάτι της άσκησης υλοποιήσατε. Η αρχική σύνδεση του πελάτη στον εξυπηρετητή θα πρέπει να είναι κάτι απλό με βάση τα παραδείγματα του μαθήματος, οπότε τα υπόλοιπα τμήματα του προγράμματος θα επηρεάσουν το μεγάλο κομμάτι της βαθμολογίας.
- Αν κάποιο κομμάτι της άσκησης σας δεν δουλεύει, αλλά δουλεύουν άλλα, σκεφτείτε να φτιάξετε τουλάχιστον ένα εκτελέσιμο για το κομμάτι της άσκησης που λειτουργεί σωστά. Είναι δύσκολο να αξιολογηθεί η άσκηση σας αν πχ σε κάποιο ερώτημα λαμβάνεται segmentation fault. Αν η άσκηση σας δεν δουλεύει σωστά, χρήσιμο είναι για την αποσφαλμάτωσή της να τυπώνετε πάντα πόσα bytes (και ποια) ΠΡΑΓΜΑΤΙΚΑ διαβάζονται/γράφονται σε κάθε read/write συνάρτηση από/σε κάθε σωλήνα ή υποδοχή. Προσοχή: Αυτό μόνο στην αποσφαλμάτωση... όχι στην παραδοτέα άσκηση, **αν το πρόγραμμά σας τρέχει σωστά.**
- Η διεργασία πατέρα του εξυπηρετητή διαβάζει σωστά ολόκληρα τα ερωτήματα των πελατών, χωρίς να κολλάει αν κάποιος δεν έχει φτάσει ακέραιο;
- Η διεργασία πατέρα του εξυπηρετητή καταλαβαίνει πότε ένας πελάτης έχει διακόψει τη σύνδεσή του και αποδεσμεύει τις πληροφορίες για αυτόν τον πελάτη? (σημαντικό)
- Γίνεται σωστά η κατανομή των εντολών στα παιδιά;
- **Όσοι παραδώσουν εργασία ατομικά (ΚΑΙ ΜΟΝΟ ΑΥΤΟΙ)** μπορούν να υλοποιήσουν ένα λιγότερο έξυπνο (και πιο απλό) τρόπο κατανομής των εντολών (**και όχι των συνδέσεων**) όπου οι εντολές θα μοιράζονται κυκλικά στις εργασίες παιδιά. Δηλαδή όσοι εργαστούν ατομικά δε χρειάζεται να ανησυχούν για το συγχρονισμό των διεργασιών-παιδιών στην ανάγνωση των εντολών.
- Εκτελούνται σωστά οι εντολές; Όλες οι κλήσεις για I/O που μπορούν να διακοπούν από σήμα, ελέγχονται για αυτό το ενδεχόμενο?
- Λαμβάνει ο πελάτης τα αποτελέσματα σωστά;

- Δουλεύει το πρόγραμμά σας αν $\text{probFail} > 0$? Μπορεί να καταλάβει ο πελάτης κάπως ότι δεν πρόκειται να λάβει απάντηση, και να συνεχίσει με άλλες εντολές, ίσως επειδή αυτή χάθηκε; Πώς; Θυμηθείτε ότι με τις τηλεγραφικές υποδοχές μηνύματα μπορούν να χαθούν και με $\text{probFail} = 0$.
- Ο πελάτης λειτουργεί σωστά αν η είσοδος του είναι από κάποιο αρχείο (με ανακατεύθυνση);
- Όλα όσα αναφέρθηκαν για χρήσεις συναρτήσεων/μεθόδων είναι ΥΠΟΧΡΕΩΤΙΚΑ. Πχ, δεν θα αξιολογηθεί άσκηση που θα χρησιμοποιεί νήματα, ή άλλες μεθόδους συντονισμού των διεργασιών στον εξυπηρετητή πέραν από σωλήνες.
- Εννοείται ότι διαφορετικές ομάδες ΔΕΝ μπορούν να έχουν την ίδια υλοποίηση οποιωνδήποτε συναρτήσεων. Κάτι τέτοιο θα θεωρηθεί αντιγραφή. Και δεν θα αποτελεί δικαιολογία αν κάποιος/κάποια έχει βγάλει το συγκεκριμένο κομμάτι κώδικα στο δίκτυο.

Extra Bonus (μέχρι 20% για υλοποίηση, 8% για απλώς λεπτομερή περιγραφή)

Τα μηνύματα στην επικοινωνία με τηλεγραφικές υποδοχές μπορούν να χαθούν. Το κεφάλαιο 20.6 του ebook περιγράφει αυτή την περίπτωση. Μπορείτε να προσθέσετε επιβεβαιώσεις (ACKs), ώστε να γίνει αξιόπιστη η επικοινωνία εξυπηρετητή-παιδιού και πελάτη; Αυτό θα πρέπει να γίνεται ΧΩΡΙΣ να κολλάει ο εξυπηρετητής-παιδί περιμένοντας λήψη ACKs, αλλά παράλληλα να μπορεί να εξυπηρετεί και άλλους πελάτες. Extra bonus, αλλά μικρότερο, θα δοθεί και σε όποιους, χωρίς να υλοποιήσουν τη σκέψη τους, γράψουν κάτι αρκετά πιο λεπτομερές (και φυσικά σωστό) από αυτό που αναφέρει το ebook στο README ως κείμενο.

Extra Bonus (μέχρι 8%)

Επιπλέον, extra bonus μέχρι 5% θα δοθεί σε όσους υλοποιήσουν τον πελάτη (ΚΑΙ ΜΟΝΟ ΑΥΤΟΝ) ώστε να μπορεί να στέλνει μαζικά παραπάνω από ένα ερωτήματα για τα οποία δεν έχει λάβει απάντηση με έξυπνο τρόπο (δηλαδή αν ο χρήστης δίνει είσοδο, να στέλνεται μήνυμα ανεξαρτήτως αν δεν έχει ληφθεί ακόμα απάντηση σε προηγούμενο ερώτημα).

Όπως πάντα, παραδίδεται ένα tar αρχείο με Makefile, README (σε greeklish), και όλα τα .h και .c αρχεία σας (όχι εκτελέσιμα). Στο tar αρχείο να έχετε δικαιώματα ΜΟΝΟ εσείς.

Όπως πάντα, η συμβουλή του διδάσκονται είναι να σχεδιάσετε κάθε λεπτομέρεια για την υλοποίηση του κώδικα (μέθοδοι, συναρτήσεις, μεταβλητές, αντίδραση σε γεγονότα) προτού γράψετε έστω μία γραμμή κώδικα. Η εξέταση της άσκησης θα περιλαμβάνει και πολλούς πελάτες και ανακατεύθυνση αρχείων, οπότε ελέγξτε καλά το πρόγραμμά σας.

Δεληγιαννάκης Αντώνιος