ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
K24: Προγραμματισμός Συστήματος
3η Προγραμματιστική Εργασία
Ημερομηνία Ανακοίνωσης: 14/5/12
Ημερομηνία Υποβολής: 1/6/12

Εισαγωγή στην Εργασία:
Ο στόχος αυτής της εργασίας είναι να δημιουργήσετε ένα απλό ωστόσο χρήσιμο ιστο–ερπετό (crawler) που θα κατεβάζει ιστοσελίδες, θα βρίσκει βασικά χαρακτηριστικά τα οποία και θα αποθηκεύει στο σύστημα αρχείου Η άσκηση θα σας βοηθήσει να εξοικειωθείτε με κλήσεις συστήματος που δημιουργούν διεργασίες, εκτελούν επιλεκτικά διαφορετικού είδους εργασίες, συνεργάζονται μέσω σωληνώσεων και τέλος συγχρονίζονται με αξιόπιστα POSIX σήματα.

Θα πρέπει να γράψετε ένα πρόγραμμα (manager) το οποίο δημιουργεί νέες διεργασίες παιδιά (children) με την βοήθεια της εντολής συστήματος fork(). Τα παιδιά εκτελούν διαφοροποιημένες εργασίες –με την βοήθεια των κλήσεων exec*()– από την διαδικασία γονιό αφού με την βοήθεια του προγράμματος wget μπορούν να 'κατεβάζουν' ιστοσελίδες τοπικά και τέλος συγχρονίζονται με την βοήθεια αξιόπιστων σημάτων.

Διαδικαστικά:
Το πρόγραμμά σας θα πρέπει να τρέχει στα μηχανήματα Linux/Unix της σχολής. Παρακολουθείτε τον ιστότοπο του μαθήματος στο URL: `www.di.uoa.gr/~ad` για επιπρόσθετες ανακοινώσεις.

- ◇ Υπεύθυνοι για την άσκηση αυτή είναι (ερωτήσεις, αξιολόγηση, βαθμολόγηση, κτλ) είναι οι κ. Νικόλας Στρατής (nicstratis-AT+di), Γιώργος Κόλλιας (grad1049-AT+di) και Κωνσταντίνος Φίλιος (konfilios@-AT+di).
- ◇ Τα προγράμματά σας θα πρέπει να γραφτούν σε *C*. Μπορείτε να χρησιμοποιήσετε και *C++* **χωρίς STL extensions** αν θέλετε. Το αποτέλεσμα της δουλείας σας θα πρέπει να τρέχει στις μηχανές του τμήματος.
- ◇ Παρακολουθείτε την ηλεκτρονική λίστα (mailman) για ερωτήσεις/απαντήσεις/διευκρινήσεις που δίνονται σχετικά με την άσκηση.

Η Είσοδος του Προγράμματος σας:
Το ιστο-ερπετό δέχεται από την γραμμή εντολής τα εξής ορίσματα:

prompt> mycrawler -f InputDataFile -u StartingURL -p NumOfProcesses -n numberOfURLs -s StatsFile

Τα ορίσματα μπορούν να εισαχθούν με οποιαδήποτε σειρά. Στην εν–λόγω γραμμή εντολής:

- mycrawler : είναι το εκτελέσιμο του βασικού σας προγράμματος.
- -f InputDataFile : είναι το όνομα του αρχείου στο οποίο υπάρχουν URLs εκκίνησης — τουλάχιστον ένα ή πιο πολλά. Σε κάθε γραμμή του αρχείου αυτού υπάρχει ένα URL π.χ.
  `www.di.uoa.gr`
  `www.di.uoa.gr/~ad/index.html`
  `www.di.uoa.gr/~k24-syspro/course.php`
  Τα παραπάνω μπαίνουν σε μια δομή 'ουράς' στο κυρίως πρόγραμμα και κάθε μία από αυτά τα URLs θα πρέπει να 'κατέβει', να αναλυθεί και να προσφέρει πιθανόν άλλα URLs που και αυτά με την σειρά τους θα πρέπει να μπουν στην ουρά και να τροφοδοτήσουν με την σειρά τους διεργασίες κατεβάσματος κοκ. Όταν χρησιμοποιείται η σημαία -f τότε δεν μπορεί να χρησιμοποιηθεί η σημαία -u και αντίστροφα.
- -u StartingURL: η εν λόγω σημαία δίνει ένα αρχικό URL από το οποίο μπορεί να ξεκινήσει το crowling. Όταν χρησιμοποιείται η σημαία -u τότε δεν μπορεί να χρησιμοποιηθεί η σημαία -f.

- -p NumOfProcesses: ο αριθμός NumOfProcesses παρέχει τον μέγιστο αριθμό από παιδιά διεργασίες που μπορούν να δημιουργηθούν και οι οποίες είναι υπεύθυνες η κάθε μία για το κατέβασμα μιας σελίδας.
- -n numberOfURLs: όταν το πρόγραμμά σας έχει επιτυχώς ολοκληρώσει το crowling από τον αριθμό numberOfURLs, όλα τα παιδιά δέχονται ένα σήμα τερματισμού, διακόπτουν ότι πιθανώς έκαναν και τερματίζουν στέλνοντας σχετικά σήματα στην αρχική διαδικασία του προγράμματος mycrawler.
- -s StatsFile: η σημαία -s ορίζει το αρχείο στο οποίο θα πρέπει να συγκεντρωθούν τα στατιστικά για όλες τις ιστοσελίδες που έχουν γίνει crowled από την εφαρμογή και τα παιδιά που έχει δημιουργήσει η εφαρμογή στην διάρκεια της εκτέλεσής της.

Στοιχεία για την Επίλυση του Προβλήματος:

Όλες οι διεργασίες συνθέτουν μια απλή ιεραρχία της οποίας η ρίζα είναι το βασικό σας πρόγραμμα. Η ρίζα του δένδρου λειτουργεί σαν ρυθμιστής της δουλειάς (manager) που πρέπει να επιτελούν οι κόμβοι φύλλα (workers).

Τα φύλλα δεν είναι απαραίτητο να είναι απλοί κόμβοι. Ίσως για να επιτευχθεί καλύτερα ο στόχος της άσκησης το κάθε παιδί του manager θα πρέπει να είναι μια ιεραρχία που επιτελεί το έρπισμα μιας συγκεκριμένης σελίδας, κατόπιν την ανάλυσή της και τέλος την μεταφορά των αποτελεσμάτων στον ρυθμιστή.

Ο βασικός ρόλος των p παιδιών είναι να παραλάβουν μια σελίδα με την βοήθεια του wget, να εξάγουν τα διάφορα στατιστικά και να παραδώσουν στον ρυθμιστή το συνολικό αποτέλεσμα της εργασίας τους. Η επικοινωνία μεταξύ ρυθμιστή και κάθε παιδιού γίνεται με την βοήθεια bidirectional pipes.

Οι ταυτόχρονοι κόμβοι που δημιουργούνται στην ιεραρχία της άσκησης θα πρέπει να δουλεύουν ανεξάρτητα. Είναι επίσης πολύ σημαντικό, σελίδες που έχουν γίνει crowled στο παρελθόν να ΜΗΝ ξαναγίνουν. Για αυτό το λόγο η διαδικασία ρυθμιστής θα πρέπει να σχεδιαστεί με τέτοιο τρόπο ώστε γρήγορα να μπορεί να απαντήσει στην ερώτηση αν έχει ήδη 'ξαναδεί' μία σελίδα (και αν κάτι τέτοιο ισχύει να μην βάλει την σελίδα στην σχετική ουρά εξυπηρέτησής του ρυθμιστή).

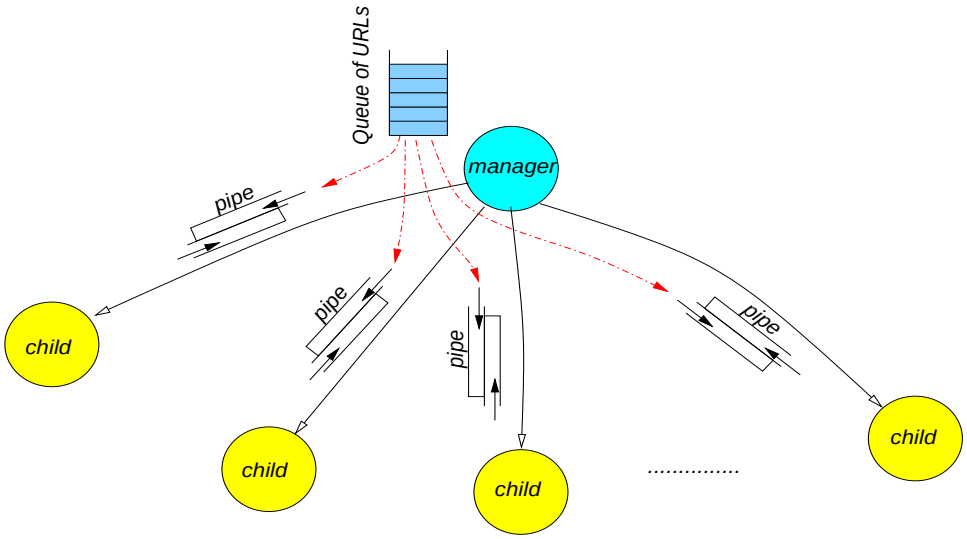
Στα πλαίσια της άσκησης θα πρέπει να κάνετε τα εξής:

1. Δημιουργία ιεραρχίας με την κλήση συστήματος fork().
2. Εκτέλεση διαφόρων προγραμμάτων από τις διεργασίες της ιεραρχίας με την χρήση των κλήσεων exec*().
3. Χρήση ενός αριθμού από κλήσεις συστήματος όπως οι getpid(), getppid(), getrusage(), exit(), κλπ. όπου κάτι τέτοιο κρίνεται απαραίτητο.
4. Συγχρονισμό διαδικασιών με αξιόπιστα σήματα όταν το ανώτατο όριο σελίδων πού έχουν γίνει crowled έχει υπερβεί την παράμετρο που ορίζεται από την σημαία -n.
5. Προφανώς σε κανένα σημείο της υλοποίησης ΔΕΝ μπορεί να χρησιμοποιηθεί η κλήση system().

Η Ιεραρχία και ο Ρόλος των Διεργασιών:

Το πρόγραμμά σας αρχικά θα πρέπει να δημιουργήσει την ιεραρχία διεργασιών παρόμοια με αυτή που φαίνεται στο Σχήμα 1.

Ο manager δημιουργεί τον κατάλληλο αριθμό από παιδιά με την βοήθεια της fork και τροφοδοτεί κάθε φορά ένα παιδί με ένα URL. Η οργάνωση και η λεπτομερής λειτουργία του κάθε παιδιού είναι θέμα σχεδιασμού του προγράμματός σας (ή των προγραμμάτων σας) και έχετε πλήρη ευελιξία όσον αφορά σε αυτό το θέμα.



Σχήμα 1: Δείγμα ιεραρχίας διεργασιών που πρέπει να δημιουργήσετε

Όταν ο μέγιστος αριθμός URLs έχει ολοκληρωθεί, ο manager στέλνει *SIGRTMIN+1* σε όλα τα παιδιά και τους ειδοποιεί ότι μόλις τελειώσουν την εργασία τους θα πρέπει να τερματίσουν αφότου στείλουν πίσω ένα *SIGRTMIN+2* σήμα. Το τελευταίο δηλώνει ότι η εργασία ενός παιδιού έχει τερματίσει.

Όταν παραληφθούν όλα τα σήματα *SIGRTMIN+2*, ο manager προχωρά στην αποθήκευση των επί μέρους αποτελεσμάτων που κάθε παιδί του έχει αποστείλει με την βοήθεια των pipes. Τα αποτελέσματα αποθηκεύονται στο ASCII αρχείο που δίνεται στην γραμμή εντολής και το mycrawler τερματίζει.

Τέλος, δεδομένα που χρησιμοποιούνται από τις διεργασίες παιδιά για να εκμαιεύσουν πληροφορίες και στατιστικά για σελίδες που έχουν κατέβει με την βοήθεια του wget διαγράφονται από τον τοπικό δίσκο πριν τα παιδιά ολοκληρώσουν την εργασία τους.

Αποτελέσματα Παιδιών και Ρυθμιστή:

Για κάθε URL το οποίο επεξεργάζεται ένα παιδί θα πρέπει να παράγονται τα παρακάτω:

1. όλα τα URLs που αναφέρονται στην υπό επεξεργασία (parsing) σελίδα. Τα URLs θα πρέπει να παραδοθούν στην διεργασία ρυθμιστής και να μπουν στην ουρά αναμονής (Σχήμα 1).
2. συνολικός αριθμός από URLs που βρέθηκαν.
3. όγκος σελίδας σε bytes.
4. τύπος του URL (δηλ. είναι directory, .html, .php, .asp, .jsp, κλπ.) που ερπετιάστηκε.
5. αριθμός από εικόνες που αναφέρονται στο εν λόγω URL.
6. χρόνος που συνολικά απαιτήθηκε από το παιδί για να προσπελάσει και να επεξεργαστεί το URL.
7. οποιαδήποτε άλλη πληροφορία που κρίνετε ότι είναι απαραίτητη και βοηθά στην καλή λειτουργία του/των προγραμμάτων σας.

Ο κόμβος ρυθμιστής θα παράγει τα εξής:

1. το αρχείο με όλα τα επιμέρους αποτελέσματα ταξινομημένα αλφαριθμητικά σύμφωνα με το URL.
2. το πιο ογκώδες URL και ο χρόνος που απαιτήθηκε για να προσπελαστεί.
3. μέσος όρος χρόνου προσπέλασης σελίδων καθώς επίσης και ο μέσος όρος όγκου URLs που ανακτήθηκαν.

Τι πρέπει να Παραδοθεί:

1. Μια σύντομη και περιεκτική εξήγηση για τις επιλογές που έχετε κάνει στο σχεδιασμό του προγράμματος σας (1-2 σελίδες ASCII κειμένου είναι αρκετές).
2. Ένα tar file με όλη σας τη δουλειά σε έναν κατάλογο που πιθανώς να φέρει το όνομά σας και θα περιέχει όλη σας τη δουλειά.

Άλλες Σημαντικές Παρατηρήσεις:

1. Οι εργασίες είναι **ατομικές.**
2. Αν και αναμένεται να συζητήσετε με φίλους και συνεργάτες το πώς θα επιχειρήσετε να δώσετε λύση στο πρόβλημα, αντιγραφή κώδικα (οποιαδήποτε μορφής) είναι κάτι που **δεν επιτρέπεται** και δεν πρέπει να γίνει. Οποιοσδήποτε βρεθεί αναμεμειγμένος σε αντιγραφή κώδικα απλά παίρνει μηδέν στο μάθημα. Αυτό ισχύει για **όλους όσους εμπλέκονται** ανεξάρτητα από το ποιος έδωσε/πήρε κλπ.
3. Όποιος υποβάλλει/δείχνει κώδικα που δεν έχει γραφτεί από την ίδια/ίδιο **μηδενίζεται** στο μάθημα.
4. Το πρόγραμμα σας θα πρέπει να τρέχει σε Ubuntu-Linux ή Solaris αλλιώς **δεν θα βαθμολογηθεί**.
5. Σε καμιά περίπτωση τα MS-Windows **δεν είναι επιλογή** πλατφόρμας για την παρουσίαση αυτής της άσκησης.