



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

Τμήμα Πληροφορικής

ΕΠΛ421 - Προγραμματισμός Συστημάτων

ΑΣΚΗΣΗ 3 – Υλοποίηση προγράμματος για την προσπέλαση δεδομένων του συστήματος αρχείων (ls –ialR)

Διδάσκων: Δημήτρης Ζεϊναλιπούρ
Υπεύθυνος εργασίας: Πάυλος Αντωνίου

Ημερομηνία Ανάθεσης: Παρασκευή 8/11/2019
Ημερομηνία Παράδοσης: Παρασκευή, 15/11/2019 (7 μέρες)
(να υποβληθεί ηλεκτρονικά μέσω Moodle)

I. Στόχος Άσκησης

Στόχος αυτής της άσκησης είναι η εξοικείωση με προχωρημένες τεχνικές προγραμματισμού στο περιβάλλον UNIX. Συγκεκριμένα, σε αυτή την άσκηση θα έχετε την δυνατότητα να κάνετε πρακτική εξάσκηση των εννοιών που έχετε διδαχθεί στις διαλέξεις όπου είδαμε τη Δομή Αρχείων, το Χαμηλού Επιπέδου I/O, Διαχείριση μέτα-πληροφοριών αρχείων <sys/stat.h>, Διαχείριση αρχείων με κλήσεις συστήματος και Προσπέλαση και Επεξεργασία καταλόγων (dirent.h) μέσω κλήσεων βιβλιοθήκης. Η υλοποίηση της άσκησης προϋποθέτει ότι έχετε κατανοήσει την δομή του συστήματος αρχείων στο UNIX καθώς επίσης την λειτουργία των I-nodes.

Η άσκηση περιλαμβάνει την υλοποίηση ενός υποσυνόλου λειτουργιών της εντολής ls, συγκεκριμένα των επιλογών (-ialR), χωρίς την κλήση της οικογένειας εντολών exec. Το πρόγραμμα σας θα πρέπει επίσης να παρέχει την δυνατότητα στους χρήστες να ορίσουν το μέγιστο βάθος διερεύνησης της αναδρομής (κάτι το οποίο δεν παρέχεται από την τυπική εντολή ls). Οι λειτουργίες του προγράμματος σας και το αναμενόμενο αποτέλεσμα περιγράφονται αναλυτικότερα στην συνέχεια.

III. Περιγραφή της εντολής ls (list segments)

Η εντολή ls παρουσιάζει το περιεχόμενο ενός καταλόγου που δίδεται σαν παράμετρος. Εάν ο χρήστης παραλείψει να δώσει παράμετρο τότε η εντολή παρουσιάζει αυτόματα το περιεχόμενο του καταλόγου «.». Επίσης ο χρήστης έχει την δυνατότητα να δώσει σαν παράμετρο πολλαπλά αρχεία. Σε αυτή την περίπτωση, θα παρουσιάσουν τα περιεχόμενα όλων των καταλόγων που ορίζονται από το χρήστη. Παρόλο που η ls έχει μια πλειάδα επιλογών, σε αυτή την άσκηση θα μας απασχολήσουν οι πιο κάτω επιλογές καθώς επίσης κάποιες εντολές οι οποίες δεν προσφέρονται αυτή τη στιγμή από την ls και οι οποίες θα περιγραφούν στη συνέχεια.

ls - list directory contents

- R list subdirectories recursively (using Depth First Search)
- l use a long listing format
- I print the index (inode) number of each file
- a include all entries (including hidden files starting with .)

Στην επόμενη εκτέλεση παρουσιάζουμε δυο παραδείγματα εκτέλεσης. Στόχος της άσκησης είναι να προσομοιώσετε αυτή την λειτουργία. Η εντολή (man ls -ial) επιστρέφει εγγράφες της μορφής:

<inode,permissions,links,owner,group,size,modification_date,name>

Ένα παράδειγμα εκτέλεσης της εντολής δίδεται πιο κάτω:

```
$ls -ialR ~/public_html/courses/epl421/  
/home/faculty/dzeina/public_html/courses/epl421/:
```

```
total 528  
/home/faculty/dzeina/public_html/courses/epl421:  
total 328  
21250971 drwxr-xr-x 9 dzeina faculty 4096 Feb 27 14:42 ./ 
```

```
/home/faculty/dzeina/public_html/courses/epl421/exercises:
```

```
total 1104  
21251006 drwxrwxrwx 2 dzeina faculty 4096 Feb 13 20:30 .  
21250971 drwxr-xr-x 9 dzeina faculty 4096 Feb 27 14:42 ..  
21266851 -rw-r--r-- 1 dzeina faculty 162480 Feb 2 01:35 ex1.pdf  
21266889 -rw-r--r-- 1 dzeina faculty 947281 Feb 14 01:06 ex2.pdf
```

*Η myls πρέπει να
διαχειρίζεται
όλους τους τύπους
UNIX Αρχείων*

```
/home/faculty/dzeina/public_html/courses/epl421/images:
```

```
total 384  
21250973 drwxr-xr-x 2 dzeina faculty 4096 Jan 27 17:41 .  
21250971 drwxr-xr-x 9 dzeina faculty 4096 Feb 27 14:42 ..  
21266573 -rw-r--r-- 1 dzeina faculty 121038 Jan 8 17:43 book.jpg  
21266572 -rw-r--r-- 1 dzeina faculty 27698 Jan 8 17:42 book-small.jpg  
21266574 -rw-r--r-- 1 dzeina faculty 75197 Jan 8 16:15 campus.jpg  
21266575 -rw-r--r-- 1 dzeina faculty 7125 Jan 13 2007 concurrency.jpg
```

*Η myls τυπώνει
εδώ τον αριθμό
των εγγραφών
στον εν λόγω
κατάλογο (εδώ 6)*

Εάν δεν δοθεί κάποια παράμετρος τότε η εντολή εκτελείται πάνω στο παρόν κατάλογο “.”:

```
$ls -ialR  
..  
total 83228  
21250731 drwxr-xr-x 2 dzeina faculty 4096 Dec 18 01:27 .  
21236693 drwxr-xr-x 12 dzeina faculty 4096 Feb 20 19:05 ..  
30064674 -rw-r--r-- 1 dzeina faculty 1881088 Dec 18 00:04 dtopk-eth-zurich.12.12.08.ppt  
30064649 -rw-r--r-- 1 dzeina faculty 1945600 Apr 4 2007 dtopk-ucy.20.03.2007.ppt  
30064675 -rw-r--r-- 1 dzeina faculty 2457088 Dec 18 00:23 dtopk-upc-barcelona.15.12.08.ppt  
30064650 -rw-r--r-- 1 dzeina faculty 5675520 Jun 14 2007 failrank.coregrid.12.06.2007.ppt  
30064655 -rw-r--r-- 1 dzeina faculty 911360 Jan 15 2008 failrank.coregrid-paris.15.1.08.ppt  
30064642 -rw-r--r-- 1 dzeina faculty 2590208 Jan 21 2008 icgrid.cygridday.21.1.08.ppt  
30064656 -rw-r--r-- 1 dzeina faculty 9758208 Jan 11 2008 microhash.cambridge.11.1.07.ppt  
30064661 -rw-r--r-- 1 dzeina faculty 9564672 Feb 14 2008 microhash.hpcl.14.2.08.ppt  
30064676 -rw-r--r-- 1 dzeina faculty 15404544 Dec 18 00:07 microhash.ibm-zurich.12.12.08.ppt  
30064657 -rw-r--r-- 1 dzeina faculty 2165959 Apr 26 2007 microhash.uci.26.04.2007.pdf  
30064651 -rw-r--r-- 1 dzeina faculty 11504128 Jun 14 2007 microhash.uci.26.04.2007.ppt  
30064659 -rw-r--r-- 1 dzeina faculty 251430 May 13 2007 micropulse.disn.11.05.2007.pdf  
30064652 -rw-r--r-- 1 dzeina faculty 540672 May 13 2007 micropulse.disn.11.05.2007.ppt  
30064658 -rw-r--r-- 1 dzeina faculty 1391104 Jul 27 2008 micropulse.hdms08.28.07.2008.ppt
```

Σημειώστε ότι γίνεται χρήση της Κατά-Βάθος Διερεύνησης (Depth-First Traversal)

III. Βοηθητικές Συναρτήσεις

Για την διεκπεραίωση αυτής της άσκησης θα χρειαστείτε κάποιες συναρτήσεις τις οποίες δεν είδαμε στο μάθημα. Συγκεκριμένα θα χρειαστείτε τις συναρτήσεις (Κεφάλαιο 6 – Stevens & Rago):

```
#include <pwd.h>
struct passwd *getpwuid(uid_t uid);
Returns: Pointer to struct passwd if OK, NULL on error
```

```
#include <grp.h>
struct group *getgrgid(gid_t gid);
Returns: Pointer to struct group if OK, NULL on error
```

α) struct passwd *getpwuid(uid_t uid);

Αντιστοιχεί το uid το οποίο περιέχετε στο i-node ενός αρχείου σε μια πιο εύληπτη μορφή (π.χ. αντί 1240 δίδει dzeina). Η εντολή αυτή ουσιαστικά ψάχνει μέσα στο /etc/passwd αρχείο. Η struct passwd περιέχει το πεδίο pw_name το οποίο είναι τύπου (char *) και το οποίο περιέχει το όνομα του χρήστη.

β) struct group *getgrgid(gid_t gid);

Αντιστοιχεί το gid το οποίο περιέχετε στο i-node ενός αρχείου σε μια πιο εύληπτη μορφή (π.χ. αντί 243 δίδει visiting). Η εντολή αυτή ουσιαστικά ψάχνει μέσα στο /etc/group αρχείο. Η struct group περιέχει το πεδίο gr_name το οποίο είναι τύπου (char *) και το οποίο περιέχει το όνομα του χρήστη.

γ) Μελετήσετε το “man stat” για να κατανοήσετε όλες τις σταθερές της βιβλιοθήκης <sys/stat.h>

IV. Ζητούμενα Άσκησης

Πιο κάτω περιγράφονται οι διαφορές επιλογές τις οποίες πρέπει να διαχειρίζεται το πρόγραμμα σας:

\$ myls: Εκτελεί αντίστοιχη λειτουργία με την εντολή **ls -ialR**

\$ myls a/ b/ c/ : Εκτελεί αντίστοιχη λειτουργία με την εντολή **ls -ialR a/ b/ c/**

\$ myls -depth 2: Εκτελεί αντίστοιχη λειτουργία με την εντολή **ls -ialR** αλλά το βάθος διερεύνησης περιορίζεται σε δυο επίπεδα.

\$ myls -follow-symbolic: ο αλγόριθμος ακολουθά και τα συμβολικά αρχεία και καταλόγους. Εάν είναι κατάλογος τότε εκτυπώνει κανονικά και τα περιεχόμενα του. Σε αυτή την περίπτωση πρέπει ο αλγόριθμος σας να ελέγχει εάν ο κατάλογος έχει ήδη εκτυπωθεί έτσι ώστε να μην τον ξανά-εκτυπώσει (και ούτε να τον επεξεργαστεί περαιτέρω). Η έλεγχος αυτός πρέπει να είναι όσο το δυνατό πιο αποδοτικός γίνεται (θα θέλαμε δηλαδή να αποφύγουμε να σαρώσουμε ολόκληρη τη λίστα των προηγούμενων μονοπατιών). Σημειώστε ότι ο αλγόριθμος διερεύνησης σε όλες τις περιπτώσεις είναι Depth-First-Search.

\$ myls -output traversal-log.txt: εκτυπώνει σε αρχείο με όνομα **traversal-log.txt** το depth-first-search μονοπάτι το οποίο ακολουθήθηκε από την εντολή myls με παρόμοιο τρόπο με το παράδειγμα που ακολουθεί

```
./
./myslides/
./myslides/epl421/
./myslides/epl421/slide1.ppt
./myslides/epl421/slide2.ppt
./myslides/epl032/
./myslides/epl032/intro.ppt
./myslides/epl001/
```

```
./exc/  
./exc2/  
./exc2/as2.doc  
./myfile.txt
```

Σημειώστε ότι η εφαρμογή σας πρέπει να υποστηρίζει οποιουσδήποτε συνδυασμούς των πιο πάνω ορισμάτων και να παράγει το αναμενόμενο αποτέλεσμα.

V. Γενικοί Κανόνες

1. Το πρόγραμμα σας πρέπει να χρησιμοποιεί τεχνικές δομημένου προγραμματισμού με την χρήση συναρτήσεων, πολλαπλών αρχείων για καλύτερη δομή του πηγαίου κώδικα, Makefile, διαχείριση λαθών συστήματος με την `error` κτλ.;
2. Για την επεξεργασία των επιλογών του προγράμματος σας θα ήταν καλό να κάνετε χρήση της βιβλιοθήκης `getopt` (για περισσότερες πληροφορίες “`man 3 getopt`”, όπου 3 δηλώνει “C library routines”)
3. Το πρόγραμμα σας πρέπει να ελαχιστοποιεί την χρήση πόρων του συστήματος (μνήμης, επεξεργαστή, στοιβάς προγράμματος, κτλ);
4. Το πρόγραμμα πρέπει να μειώνει όσο το δυνατό περισσότερο τον χρόνο διεκπεραίωσης της εντολής.
5. Συμβουλευτείτε τις οδηγίες του ΕΠΛ232 για συγγραφή των σχετικών ελέγχων μονάδων, διάσπαση του προγράμματος σε πολλαπλά αρχεία, τεκμηρίωση, `makefile`, κτλ.

Καλή Επιτυχία !