

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
4η Εργασία - Τμήμα: Περιττών Αριθμών Μητρώου
Κ22: Λειτουργικά Συστήματα – Χειμερινό Εξάμηνο '24
Ημερομηνία Ανακοίνωσης: Σάββατο 04 Ιανουαρίου 2025
Ημερομηνία Υποβολής: Παρασκευή 14 Φεβρουαρίου 2025 Ώρα 23:55

Εισαγωγή στην Εργασία:

Ο στόχος αυτής της εργασίας είναι να δημιουργήσετε ένα πρόγραμμα συστήματος, με την ονομασία `myz`, που επιπεδοποιεί λογικές ιεραρχίες καταλόγων και αρχείων στο LINUX. Το αρχείο που δημιουργείται έχει postfix `.myz` και μπορεί να διαθέτει τα περιεχόμενα των απλών αρχείων στην ιεραρχία σε συμπτυκνωμένη μορφή με την βοήθεια του προγράμματος `gzip` όπως επίσης και όλες τις απαραίτητες πληροφορίες σχετικά με την δομή της λογικής ιεραρχίας στο LINUX. Το πρόγραμμα σας θα πρέπει να δέχεται σαν παραμέτρους εισόδου σημαίες και ονόματα/λίστες αρχείων/καταλόγων που θα πρέπει να αποθηκευτούν στο υπό διαμόρφωση `.myz` αρχείο.

Το πρόγραμμα σας θα πρέπει με επιτυχία να «ανακτά» είτε ολόκληρη ή μέρος της ιεραρχίας καταλόγων/αρχείων από το επιπεδοποιημένο `.myz` αρχείο. Θα πρέπει να τονιστεί ότι το `myz` θα πρέπει να διαχειρίζεται όχι μόνο καταλόγους και αρχεία αλλά και συνδέσμους αρχείων. Το ζητούμενο πρόγραμμα θα πρέπει να δουλεύει με παρεμφερή τρόπο όπως εκείνος των υπηρεσιών `zip`, `tar`, `rar` κλπ.

Διεπαφή του `myz` και Βασικές Λειτουργίες:

Οι σημαίες της γραμμής κλήσης καθορίζουν την λειτουργικότητα του `myz`. Σε περίπτωση που δοθούν λάθος παράμετροι, το πρόγραμμα σας θα πρέπει να δίνει κάποια ένδειξη λάθους και να τερματίζει. Παρακάτω, περιγράφουμε την διεπαφή του `myz` και την αναμενόμενη λειτουργία των επιμέρους σημαιών:

```
>>> myz {-c|-a|-x|-m|-d|-p|-j} <archive-file> <list-of-files/dirs>
```

Οι σημαίες της γραμμής εντολής δουλεύουν ως εξής:

- c Αρχειοθέτηση των οντοτήτων συστήματος αρχείου που παρέχονται από την `<list-of-files/dirs>` στο αρχείο `<archive-file>`. Το τελευταίο αρχείο μπορεί να έχει επίθεμα `.myz`. Κατάλογοι και αρχεία που είναι φωλιασμένα στην λίστα `<list-of-files/ dirs>` αποθηκεύονται με αναδρομικό τρόπο στο `<archive-file>`.
- a Προσθήκη των εγγράφων και καταλόγων στην `<list-of-files/dirs>` στον πιθανά ήδη υπάρχον αρχείο `<archive-file>`. Αν υπάρχουν φωλιασμένοι κατάλογοι στο `<list-of-files/dirs>` περιλαμβάνονται αναδρομικά μαζί με όλο το περιεχόμενο τους στο `<archive-file>`.
- x Εξαγωγή όλων των εγγράφων/καταλόγων που βρίσκονται αποθηκευμένοι στο `<archive-file>` στο τωρινό κατάλογο εργασίας (current working directory).
- j Τα έγγραφα πριν την αρχειοθέτηση στο `<archive-file>` θα πρέπει να συμπιεστούν με `gzip`.
- d Διαγραφή των οντοτήτων που αναφέρονται στη λίστα `<list-of-files/dirs>` από το υπάρχον αρχείο επιπεδοποίησης `<archive-file>`.
- m Εκτύπωση μεταπληροφοριών (owner, group, access rights) για όλες τις οντότητες που είναι αποθηκευμένες στο `<archive-file>`.
- q Επερώτηση για το αν η όχι τα στοιχεία της λίστας `<list-of-files/dirs>` βρίσκονται στο αρχείο `<archive-file>`. Για κάθε στοιχείο της `<list-of-files/dirs>` θα παρέχεται μια θετική ή μια αρνητική απάντηση.
- p Εκτύπωση της ιεραρχίας των καταλόγων και των αρχείων που βρίσκονται στο `<archive-file>` με τρόπο που να είναι εύκολα αντιληπτός/αναγνώσιμος.

Μερικές παρατηρήσεις:

- Το πρόγραμμα σας δεν είναι απαραίτητο να χρησιμοποιεί την στάνταρντ είσοδο/έξοδο με οποιοδήποτε άλλο τρόπο πλην αυτού που περιγράφεται παραπάνω.
- Η διαδικασία ανάκτησης περιλαμβάνει μόνο αρχεία ή/και καταλόγους που ο χρήστης παραθέτει στην γραμμή εκτέλεσης του προγράμματος με την χρήση της παραμέτρου/λίστας <list-of-files/dirs>. Αν δεν δίνεται καμία τέτοια λίστα, με την επίκληση της σημαίας -x, όλες οι αποθηκευμένες οντότητες «ανα-δημιουργούνται» στο κατάλογο που διενεργείται η ανάκτηση.
- Η σημαία -j μπορεί να χρησιμοποιηθεί μόνο μαζί με την σημαία -c ή την -a.
- Στην διαδικασία προσθήκης με την σημαία -a αλλά και διαγραφής με την σημαία -d, θα πρέπει να γίνουν οι ελάχιστες πιθανόν αλλαγές στο <archive-file>. Σε κάθε περίπτωση θα πρέπει να αποφευχθεί η συνολική αναπαραγωγή του <archive-file>.
- Όταν γίνεται ανάκληση αρχείων, η τελική μορφή των αρχείων στην (λογική) ιεραρχία θα είναι χωρίς συμπίεση (ανεξάρτητα αν είχε υπάρξει συμπίεση περιεχομένου στα σχετικά αρχεία).

Διαδικαστικά:

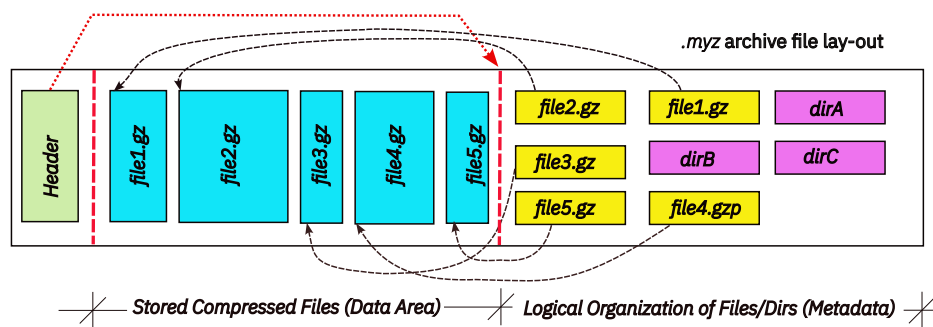
- Υπεύθυνοι για την άσκηση αυτή (ερωτήσεις, αξιολόγηση, βαθμολόγηση κλπ.) είναι ο Δρ. Σαράντης Πασκαλής paskalis+AT-di, ο κ. Νίκος Λαζαρόπουλος niklaz+AT-di, και ο κ. Δημήτρης Ροντογιάννης dronto+AT-di.
- Το πρόγραμμα σας θα πρέπει να γραφτεί σε C (ή C++ αν θέλετε αλλά χωρίς τη χρήση STL/Templates) και να τρέχει στα LINUX workstations του τμήματος.
- Ο πηγαίος κώδικας σας (source code) πρέπει να αποτελείται από **τουλάχιστον δυο** (και κατά προτίμηση πιο πολλά) διαφορετικά αρχεία και θα πρέπει *απαραιτήτως να γίνεται χρήση* separate compilation .
- Παρακολουθείτε την ιστοσελίδα του μαθήματος <https://www.alexdelis.eu/k22/> για επιπρόσθετες ανακοινώσεις.

Ανάπτυξη του Προγράμματος σας:

Η αποθήκευσή της λίστας των αρχείων/καταλόγων στο .myz αρχείο, θα πρέπει να γίνει με ένα τρόπο που βοηθά την ανάκτηση της πληροφορίας και να είναι συμπαγής. Εκτός από τα απλά αρχεία, το .myz αρχείο θα πρέπει να αποθηκεύει με ορθό τρόπο την λογική ιεράρχησή καταλόγων αλλά και τις μεταπληροφορίες για όλες τις αποθηκευμένες οντότητες συστήματος αρχείου όπως κάτοχο (owner), ομάδα (group), δικαιώματα προσπέλασης (access rights), timestamps δημιουργίας/ αλλαγής/ προσπέλασης, κλπ.

Το Σχήμα 1 προτείνει μια γενική διάταξη για τα επιπεδοποιημένα αρχεία που θα παράγει το myz. Μπορείτε να επεκτείνετε η και να αλλάξετε την παραπάνω δομή όπως εσείς θεωρείτε καλύτερα ώστε να επιτύχετε συμπαγή οργάνωση και γρήγορη λειτουργία (ακόμα και όταν μεγάλες σε όγκο και μορφή ιεραρχίες καταλόγων πρέπει να επιπεδοποιηθούν). Θα πρέπει ωστόσο να εξηγήσετε και να δικαιολογήσετε τις σχεδιαστικές επιλογές σας.

Η προτεινόμενη οργάνωση του .myz αρχείου αποτελείται από τρία τμήματα: μια επικεφαλίδα, μια σειρά από τα (συμπίεμένα) αρχεία και μια δομή που να αναπαριστά με ακρίβεια την λογική ιεραρχία και τις μεταπληροφορίες των οντοτήτων που αποθηκεύονται. Όλη η παραπάνω οργάνωση αποθηκεύεται στο .myz αρχείο στο δίσκο. Πιο συγκεκριμένα: 1) Η επικεφαλίδα μπορεί να βαστάει οποιαδήποτε στοιχεία που περιγράφουν την φύση του .myz αρχείου όπως π.χ. μέγεθος σε bytes αλλά και ένα δείκτη (δηλ. αριθμό byte) πέρα από τον οποίο αποθηκεύεται η αναπαράσταση της δομής της ιεραρχίας (δηλ. μεταδεδομένα). Το μέγεθος της επικεφαλίδας παραμένει σταθερό και είναι γνωστό εξ' αρχής. 2) Τα (συμπίεμένα) αρχεία αποθηκεύονται το ένα μετά το άλλο στο δεύτερο τμήμα του .myz αρχείου. 3) Το τρίτο και τελευταίο κομμάτι του .myz αρχείου είναι ουσιαστικά το «λεξικό» το οποίο διαθέτει όλες της πληροφορίες ιεράρχησης της <list-of-files/dirs> και τις μεταπληροφορίες των επιμέρους οντοτήτων (αρχεία, κατάλογοι, δεσμοί).



Σχήμα 1: Παράδειγμα λογικής οργάνωσης της δομής του αρχείου .myz

Θα πρέπει να τονιστεί ότι η <list-of-files/dirs> μπορεί να περιέχει υποκαταλόγους πολλαπλών επιπέδων των οποίων το βάθος δεν είναι γνωστό εκ των προτέρων. Το myz θα πρέπει αναδρομικά να καλύπτει όλη την ιεραρχία καταλόγων όπως επίσης και φωλιασμένων αρχείων και δεσμών. Τέλος, όταν το πρόγραμμα σας επικαλείται με σημαία -j θα πρέπει για κάθε αρχείο με περιεχόμενο (ASCII ή binary) να γίνει κλήσης της gzip. Αυτό μπορεί να γίνεται με ταυτόχρονη επίκληση fork()/exec*() κατά τα γνωστά. Στην διάρκεια της ανάκτησης η αντίστροφη διαδικασία θα πρέπει να ακολουθηθεί.

Το πρόγραμμα σας θα μπορεί να διαβάσει πληροφορίες για i-nodes χρησιμοποιώντας την κλήση stat (χάνετε man -S 2 stat). Επίσης θα φανούν χρήσιμες οι παρακάτω κλήσεις lstat, readdir, opendir, readlink, fchdir, chdir, getcwd, open, close, read, write, κλπ.

Τι πρέπει να Παραδοθεί:

1. Μια σύντομη και περιεκτική εξήγηση για τις επιλογές που έχετε κάνει στο σχεδιασμό του προγράμματος σας (3-4 σελίδες σε ένα PDF αρχείο).
2. Ένα Makefile (που να μπορεί να χρησιμοποιηθεί για να γίνει αυτόματα το compile του προγράμματος σας).
3. Ένα zip/7z αρχείο με όλη σας τη δουλειά σε έναν κατάλογο που πιθανώς να φέρει το όνομα σας και θα περιέχει όλη σας τη δουλειά δηλ. source files, header files, output files (αν υπάρχουν) και οτιδήποτε άλλο χρειάζεται.

Βαθμολόγηση Προγραμματιστικής Άσκησης:

Σημεία Αξιολόγησης Άσκησης	Ποσοστό Βαθμού (0-100)
Ποιότητα στην Οργάνωση Κώδικα & Modularity	10%
Επαρκής/Κατανοητός Σχολιασμός Κώδικα	05%
Υλοποίηση Ιεραρχίας (-c) και Μεταδεδομένων	30%
Υλοποίηση σκληρών/μαλακών δεσμών σε αρχεία	05%
Υλοποίηση μερικής/συνολικής εξαγωγής οντοτήτων (-x)	10%
Υλοποίηση μερικής διαγραφής οντοτήτων (-d)	10%
Υλοποίηση συμπίεσης αρχείων	05%
Πληροφορίες, επερωτήσεις και Εκτύπωση της ιεραρχίας (-m/-q/-p)	10%
Υλοποίηση επιλεκτικής προσθήκης οντοτήτων (-a)	15%

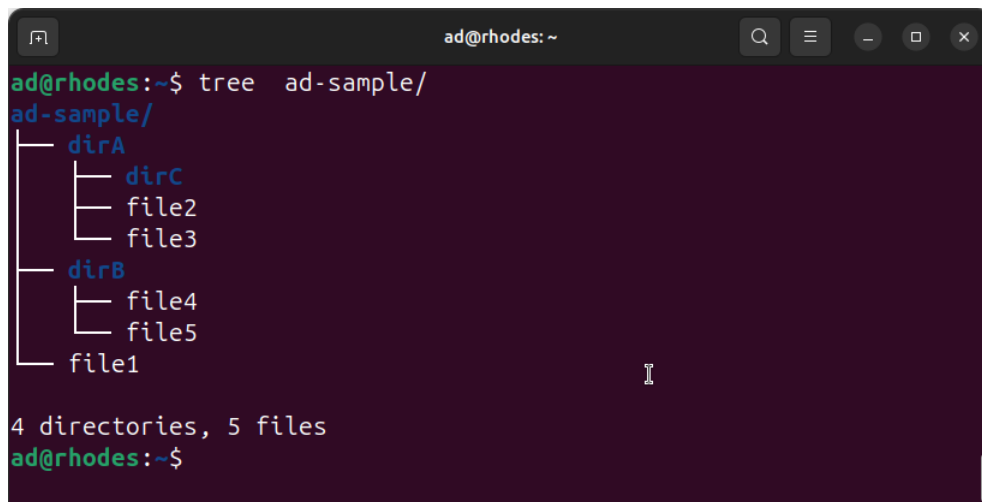
Άλλες Σημαντικές Παρατηρήσεις:

1. Η εργασία μπορεί να γίνει από ομάδες των πολύ 2 ατόμων. Αν θέλετε μπορείτε να την κάνετε και ατομικά.
2. Το πρόγραμμα σας θα πρέπει να τρέχει στα LINUX συστήματα του τμήματος αλλιώς δεν μπορεί να βαθμολογηθεί.

3. Αν και αναμένεται να συζητήσετε με φίλους και συνεργάτες το πως θα επιχειρήσετε να δώσετε λύση στο πρόβλημα, αντιγραφή κώδικα (οποιαδήποτε μορφής) είναι κάτι που δεν επιτρέπεται και δεν πρέπει να γίνει. Οποιοσδήποτε βρεθεί αναμειγμένος σε αντιγραφή κώδικά απλά παίρνει μηδέν στο μάθημα. Αυτό ισχύει για όσους εμπλέκονται ανεξάρτητα από το ποιος έδωσε/πήρε κλπ.
4. Το παραπάνω επίσης ισχύει αν διαπιστωθεί *έστω και μερική άγνοια* του κώδικα που έχετε υποβάλει ή *άπλα υπάρχει υποψία* ότι ο κώδικας είναι προϊόν συναλλαγής με τρίτο/-α άτομο/α ή με συστήματα αυτόματης παραγωγής λογισμικού ή με αποθηκευτήρια (repositories) οποιασδήποτε μορφής.
5. Αναμένουμε ότι όποιος/-οι υποβάλουν την εν λόγω άσκηση θα πρέπει να έχει πλήρη γνώση και δυνατότητα εξήγησης του κώδικα. Αδυναμία σε αυτό το σημείο οδηγεί σε μηδενισμό στην άσκηση.
6. Σε καμιά περίπτωση τα Windows δεν είναι επιλέξιμη πλατφόρμα για την υλοποίηση αυτής της άσκησης.

Appendix: Μια μέθοδος Οργάνωσης των Μεταδεδομένων σε .myz Αρχεία

Το Σχήμα 2 παρουσιάζει μια λογική ιεραρχία κομμάτι της οποίας αποθηκεύεται σε .myz αρχείο. Ενώ βρισκόμαστε στο αρχείο ad-sample επιθυμούμε να επιτεδοποιήσουμε μαζί το αρχείο file1 και τους καταλόγους dirA και dirB. Εμφανώς τμήματα της ιεραρχίας πάνω από το ad-sample παραμένουν εκτός αποθήκευσης.



```
ad@rhodes:~$ tree ad-sample/
ad-sample/
├── dirA
│   ├── dirC
│   │   ├── file2
│   │   └── file3
│   └── dirB
│       ├── file4
│       └── file5
└── file1

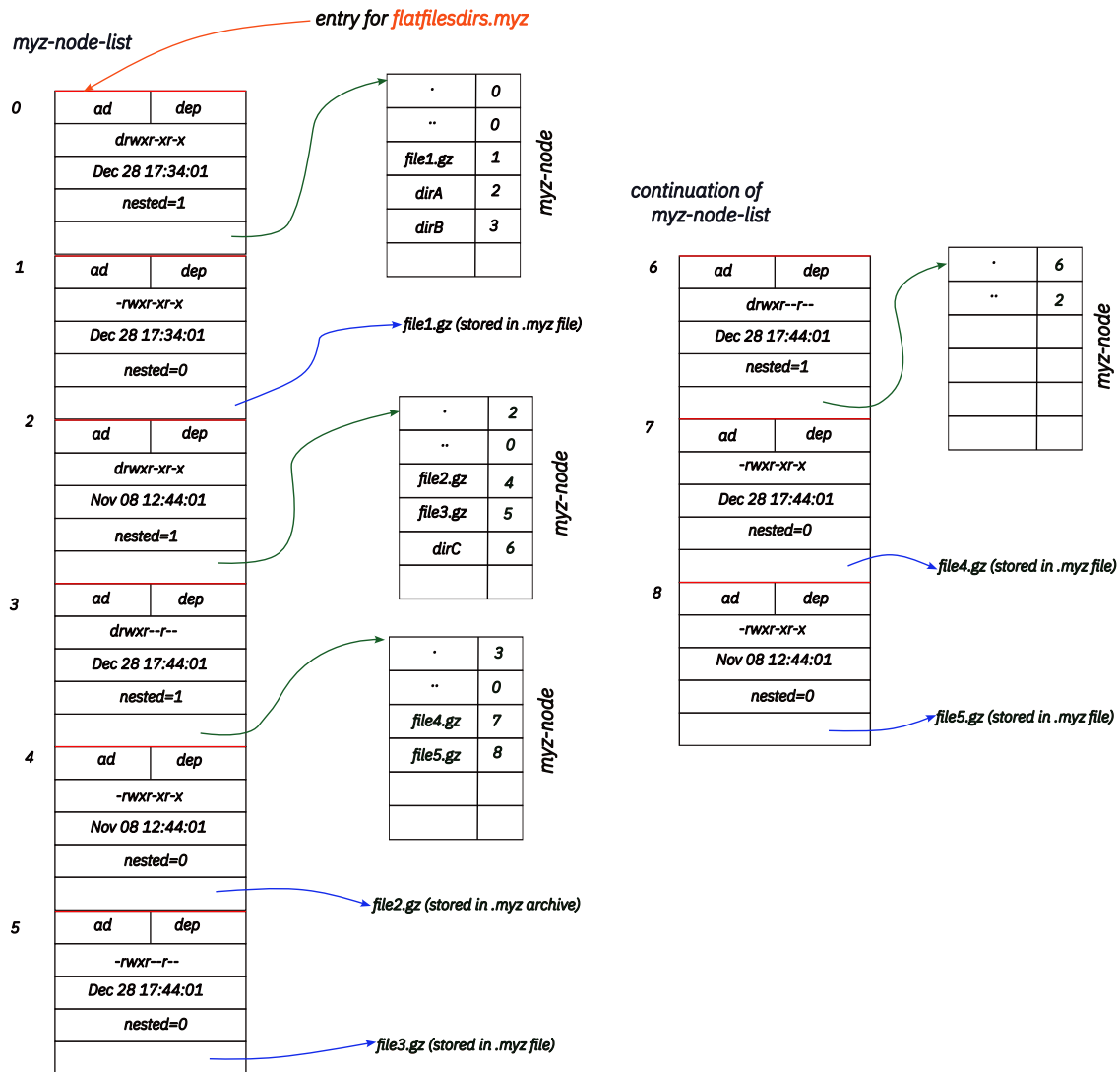
4 directories, 5 files
ad@rhodes:~$
```

Σχήμα 2: Παράδειγμα τμήματος ιεραρχίας που αποθηκεύεται σε σε αρχείο .myz

Αν από το κατάλογο /home/ad/ad-sample/ εκτελέσουμε την εντολή:

```
>>> /home/ad/bin/myz -c flatfilesdirs.myz file1 dirA dirB
```

τότε το Σχήμα 3 παρουσιάζει μια *προτεινόμενη* οργάνωση μεταδεδομένων για τα αρχεία και καταλόγους που αποθηκεύονται στο υπό σύνθεσή flatfilesdirs.myz αρχείο. Η προτεινόμενη οργάνωση βασίζεται σε μια απλουστευμένη μορφή των i-node-list και i-nodes του LINUX, και με παρόμοιο τρόπο αποτελείται από μία myz-node-list και προσαρτημένους myz-node(ς). Και οι δύο αυτές οντότητες μπορούν να δημιουργούνται με βάση μπλοκ τα όποια έχουν σταθερό μέγεθος (ας πούμε 512 ή 1024 Bytes). Με την παραπάνω οργάνωση των μετα-δεδομένων, μερική αναδιοργάνωση του .myz αρχείου δεν οδηγεί στην συνολική του ανακατασκευή.



Σχήμα 3: Παράδειγμα οργάνωσης της μεταδεδομένων του αρχείου .myz με την βοήθεια μιας myz-node-list και σχετικών myz-node(ς) στο έγγραφο αρχειοθέτησης.