

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**  
**Τμήμα Πληροφορικής & Τηλεπικοινωνιών**  
**4η Εργασία – Λειτουργικά Συστήματα – Χειμερινό Εξάμηνο '08**  
**Ημερομηνία Ανακοίνωσης: Τετάρτη, 14 Ιανουαρίου**  
**Ημερομηνία Υποβολής: Δευτέρα, 16 Μαρτίου και Ωρα 10:00πμ**

### **Εισαγωγή στην Εργασία:**

Στα πλαίσια αυτής της άσκησης θα πρέπει να γράψετε μια εφαρμογή που ονομάζεται `mybaya` και έχει συμπεριφορά όμοια με αυτή των προγραμμάτων `tar` και `zip` στο Solaris/Linux. Το πρόγραμμα σας που θα ονομάζεται `mybaya` θα δέχεται σαν παραμέτρους μια σειρά από αρχεία και καταλόγους, τα οποία θα αποθηκεύει σε ένα μόνο αρχείο με επέκταση. Ο τρόπος αποθήκευσης θα πρέπει να είναι τέτοιος, ώστε να μπορούν να ανακατασκευαστούν τα αρχεία και οι κατάλογοι από αυτό καθώς και τα χαρακτηριστικά τους.

### **Διαδικαστικά:**

- ◊ Τα προγράμματα σας θα πρέπει να γραφτούν σε `C` (ή `C++` χωρίς όμως `STL extensions`) και να τρέχουν στις μηχανές Unix (Solaris ή Linux) του τμήματος.
- ◊ Για την υλοποίηση θα πρέπει να χρησιμοποιηθούν κλήσεις συστήματος που δίνουν πληροφορίες καταλόγων και αρχείων. Πληροφορίες για τις απαιτούμενες κλήσεις συστήματος μπορούν να βρεθούν στη τεκμηρίωση των συστημάτων Solaris/Linux (δηλ. σχετικές `man pages`) και σε πληροφορίες που είναι διαθέσιμες από την σελίδα του μαθήματος.
- ◊ Παρακολουθήστε την ιστοσελίδα του μαθήματος για επιπρόσθετες ανακοινώσεις στο URL: [www.di.uoa.gr/~ad/OSdir/index.html](http://www.di.uoa.gr/~ad/OSdir/index.html) όπως και την λίστα `mailman` του μαθήματος.
- ◊ Υπεύθυνοι για την άσκηση (δηλ. ερωτήσεις, αξιολόγηση, βαθμολογία) είναι οι κ. Ιωάννης Πόγκας ([grad0948@di.uoa.gr](mailto:grad0948@di.uoa.gr)) και Χαράλαμπος Νικολάου ([charnik@di.uoa.gr](mailto:charnik@di.uoa.gr)).

### **Διατύπωση του Προβλήματος:**

Το πρόγραμμα σας με όνομα `mybaya` θα αναγνωρίζει τα παρακάτω ορίσματα:

`mybaya {-c|-a|-x|-m|-p|-j} <archive-file> <file/directory list>`

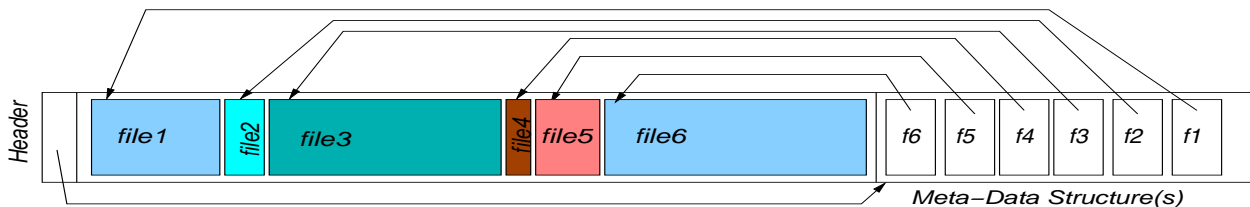
- c Αρχειοθέτηση των εγγράφων της λίστας `<file/directory-list>` στο αρχείο `<archive-file>` (δηλ. δημιουργία του αρχείου `<archive-file>`). Αν υπάρχουν κατάλογοι στην `<file/directory-list>` τότε όλα τα περιεχόμενα αρχεία και υποκατάλογοι αποθηκεύονται αναδρομικά στο `<archive-file>`.
- a Προσθήκη των εγγράφων της λίστας `<file/directory list>` στο ήδη υπάρχον αρχείο `<archive-file>`. Αν υπάρχουν κατάλογοι στην `<file/directory list>` περιλαμβάνονται αναδρομικά (μαζί με το περιεχόμενο τους) στο `<archive-file>`.
- x Εξαγωγή όλων των εγγράφων/καταλόγων που υπάρχουν αρχειοθετημένα στο αρχείο `<archive-file>`.
- m Εκτύπωση μεταπληροφοριών (owner, group, rights) για τους φακέλους/καταλόγους που είναι αρχειοθετημένοι στο `<archive-file>`.
- p Εκτύπωση της ιεραρχίας των φακέλων και καταλόγων που είναι αρχειοθετημένα στο `<archive-file>` με μια μορφή που είναι εύκολα αντιληπτή.
- j Αρχειοθέτηση των εγγράφων της λίστας `<file/directory list>` σε συμπιεσμένη μορφή στο αρχείο `<archive-file>`.

### **Μερικές παρατηρήσεις:**

1. Σε αντίθεση με την συνήθη συμπεριφορά των `tar/zip`, το `mybaya` δεν χρειάζεται να παίρνει περισσότερα ορίσματα από τα παραπάνω και δεν θα χρησιμοποιεί σε καμία περίπτωση την προκαθορισμένη είσοδο/έξοδο.

2. Η διαδικασία της ανάκτησης πρέπει να περιλαμβάνει μόνο τα αρχεία που δίνονται στη λίστα αρχείων, εκτός από την περίπτωση που η λίστα είναι κενή που υπονοεί το σύνολο των αρχείων ή καταλόγων.
3. Τονίζεται ότι η επιλογή `-j` μπορεί να χρησιμοποιηθεί μόνο σε συνδυασμό με την επιλογή `-c` ή την επιλογή `-a`.
4. Αν γίνεται προσθήκη φακέλων/καταλόγων σε υπάρχον `.baya` αρχείο, η εν λόγω προσθήκη μπορεί να γίνει με ή χωρίς συμπίεση αρχείων.
5. Ανεξάρτητα από το γεγονός ότι τα αρχεία έχουν αρχικά αρχειοθετηθεί σε συμπιεσμένη ή όχι μορφή, όταν γίνεται η ανάκληση δεδομένων του `mybaya` με σημαία `-x` όλα τα αρχεία τελικά ανακτώνται χωρίς συμπίεση.

Η αρχειοθέτηση της λίστας των αρχείων θα πρέπει να γίνεται με τέτοιο τρόπο που να είναι δυνατή η (εύκολη) ανάκτησή τους. Εκτός από το περιεχόμενό τους θα πρέπει να μπορεί να ανακτηθούν και οι μεταπληροφορίες τους, όπως ο κάτοχός τους (**owner**), η ομάδα τους (**group**) και τα δικαιώματά τους (**rights**). Προτείνουμε μια γενική διάταξη όπως αυτή του Σχήματος 1 για την οργάνωση των `.baya` αρχείων. Καλείστε να επεκτείνετε ή να τροποποιήσετε όπως νομίζετε, αλλά για συγκεκριμένους λόγους που θα πρέπει να εξηγήσετε.



Σχήμα 1: Πιθανή οργάνωση ενός `.baya` αρχείου.

Στην παραπάνω οργάνωση το κυρίως σώμα του `.baya` αρχείου περιέχει όλους τους αρχειοθετημένους φακέλους και καταλόγους (δηλ. archived data). Όλα τα αρχεία γράφονται το ένα μετά το άλλο. Στο τέλος του σώματος του αρχείου, το «λεξικό» (dictionary) περιέχει μία λίστα από εγγραφές, μία για κάθε αρχείο/κατάλογο. Κάθε μία τέτοια εγγραφή περιέχει το όνομά του, το μέγεθός του και την θέση που ξεκινούν τα δεδομένα του αρχείου/καταλόγου στο τμήμα των αρχειοθετημένων δεδομένων. Τέλος, η κεφαλίδα (header) του αρχείου `.baya` η οποία είναι σταθερού μεγέθους, δίνει (ανάμεσα σε πιθανόν άλλες πληροφορίες) την αρχική θέση του λεξικού στο αρχείο.

Θα πρέπει να τονίσουμε πως η λίστα αρχείων μπορεί να περιέχει καταλόγους, οπότε όλα τα περιεχόμενα σε αυτούς θα πρέπει να αρχειοθετηθούν. Σε περίπτωση που ένας κατάλογος περιέχει κάποιον υποκατάλογο, θα πρέπει να αρχειοθετηθεί και αυτός. Η αρχειοθέτηση καταλόγων συνεχίζεται με αυτό τον τρόπο αναδρομικά για κάθε υποκατάλογο. Αντίστοιχα, στην περίπτωση ανάκτησης αν ζητηθεί κάποιος κατάλογος, πρέπει να ανακτηθεί όλη η δομή με ρίζα τον κατάλογο και να αναπαρασταθεί κανονικά (δηλ. να δημιουργηθούν και οι υπο-κατάλογοι).

Στην περίπτωση της συμπίεσης, θα πρέπει αρχικά να εφαρμόσετε τον αλγόριθμο συμπίεσης *Lempel-Ziv-Welch (LZW)* [1] σε κάθε αρχείο της λίστας παραμέτρων και στη συνέχεια να αποθηκεύσετε το αποτέλεσμα με τον τρόπο που προαναφέρθηκε (ή με το τρόπο που τελικά εσείς επιλέξετε να δομήσετε το `.baya` αρχείο). Στην περίπτωση των καταλόγων η συμπίεση εφαρμόζεται μόνο στα αρχεία που περιέχει. Για δική σας διευκόλυνση σας μπορείτε να χρησιμοποιήσετε την υλοποίηση του προγράμματος (`compress`) που σας δίνουμε στην σελίδα του μαθήματος και η οποία χρησιμοποιεί μια παραλλαγή του (*LZW*). Για κάθε έγγραφο που πρόκειται να συμπίεστεί, το πρόγραμμα σας θα δημιουργεί μία νέα διεργασία με την βοήθεια της `fork()`. Η τελευταία θα καλεί το `compress` με είσοδο το έγγραφο και θα παράγει ένα αρχείο με κατάληξη `.Z`, το οποίο θα αποθηκεύεται στο `<archive-file>`. Ανάλογα με την σχεδίαση που τελικά θα επιλέξετε μπορούν να υπάρχουν πολλαπλές

ταυτόχρονες κλήσεις *fork()* (όταν μια λίστα αρχειοθέτησης —περιέχει πάνω από δυο αρχεία). Η αντίστροφη διαδικασία λαμβάνει χώρα κατά την αποσυμπίεση του αρχείου από το `<archive-file>`.

### Τι πρέπει να παραδοθεί:

1. Όλη η δουλειά σας σε ένα tar-file που να περιέχει όλα τα source files, header files, output files, makefile.
2. Μια σύντομη περιγραφή (2-3 σελίδες) για τις επιλογές που κάνατε στο σχεδιασμό της άσκησης. Επίσης πρέπει να αναφέρετε σε ποιο λειτουργικό σύστημα δοκιμάστηκε η δουλειά σας (δηλ. Solaris ή Linux ).
3. Ένα (ή πιο πολλά) Makefile που χρησιμοποιήσατε για να γίνουν αυτόματα compile τα προγράμματα σας.

### Άλλες Σημαντικές Παρατηρήσεις:

1. Οι εργασίες είναι ατομικές.
2. 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 αναμεμιγμένος σε αντιγραφή κώδικα απλά παίρνει μηδέν στο μάθημα. Αυτό ισχύει για **όλους όσους εμπλέκονται** ανεξάρτητα από το ποιος έδωσε/πήρε κλπ.
3. Το πρόγραμμα σας θα πρέπει να τρέχει σε Solaris ή Linux αλλιώς **δεν θα βαθμολογηθεί**.
4. Σε καμιά περίπτωση τα MS-Windows **δεν είναι επιλογή** πλατφόρμας για την παρουσίαση αυτής της άσκησης.

### Αναφορές

- [1] T. A. Welch. A Technique for High-performance Data Compression. *IEEE Computer*, 17(6):8–19, 1984.