



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης  
Τμήμα Πληροφορικής  
2012-2013

Λειτουργικά Συστήματα  
Εργασία 1<sup>η</sup> | Δημιουργία Κέλυφους σε Linux OS

Συμμετέχοντες:

Όνομα	AEM	email
Νεστορίδης Αντώνης	2078	antonest@csd.auth.gr
Οικονομίδης Γεώργιος	2080	georgido@csd.auth.gr
Πίσκας Γεώργιος	2087	gpiskasv@csd.auth.gr
Σιώζος-Δρόσος Σωκράτης	2134	siozosdr@csd.auth.gr

Σκοπός της εργασίας αυτής είναι η δημιουργία ενός κελύφους, για την διεπαφή μεταξύ του χρήστη και του πυρήνα σε ένα λειτουργικό σύστημα LINUX. Ακολουθεί μια παρουσίαση του κελύφους αυτού όπου θα παρουσιαστούν τα εξής χαρακτηριστικά:

- Δομή του κελύφους
- Λειτουργικότητα και δυνατότητες
- Περιορισμοί

Η γενική φιλοσοφία του προγράμματος είναι πώς θα λειτουργεί μέσα σε μια συνεχή επανάληψη στην οποία ο χρήστης θα δίνει συμβολοσειρές, οι οποίες αντιπροσωπεύουν εντολές και ορίσματα. Η επανάληψη αυτή είναι χωρισμένη σε τρία τμήματα τα οποία αφορούν:

1. Την είσοδο του χρήστη (συμβολοσειρά input)
2. Την αποκωδικοποίηση και επεξεργασία της εισόδου αυτής (parsing)
3. Την εκτέλεση των επεξεργασμένων εντολών

Η επανάληψη αυτή θα λειτουργεί ασαμάτητα εκτός και αν ο χρήστης επιθυμεί την έξοδο του από το πρόγραμμα και δώσει την εντολή exit.

Για να δοθεί input στο πρόγραμμα, είτε διαβάζεται από το χρήστη (πληκτρολόγιο), είτε δοθείσας μιας εντολής hist # φορτώνεται μια προηγούμενη εντολή, η οποία εκτελέστηκε προηγουμένως. Η hist είναι μια built-in εντολή και θα περιγραφεί παρακάτω.

Έχοντας αποκτήσει input από το πρώτο τμήμα, προχωράμε στο parsing. Στις απλές περιπτώσεις εισόδου, δηλαδή σε είσοδο της μορφής «cmd arg1 arg2 arg3 arg4» χωρίς χρήση wildcard, redirection και piping το πρόγραμμα εκτελεί απλουστευμένο parsing. Όταν υπάρχουν οι παραπάνω περιπτώσεις (wildcard, redirection, piping) απαιτείται πολυπλοκότερη προεπεξεργασία ή και εκτέλεση κάποιου ξεχωριστού κομματιού κώδικα στο τρίτο τμήμα της επανάληψης (piping). Ο έλεγχος για το αν η εντολή πρέπει να εκτελεσθεί στο παρασκήνιο γίνεται στην αρχή του parsing, σε κάθε περίπτωση. Η παραπάνω επεξεργασία, είναι υλοποιημένη σε πολλές συναρτήσεις για να είναι το πρόγραμμα ευανάγνωστο.

Το τρίτο τμήμα αφορά την εκτέλεση της επεξεργασμένης εισόδου. Ελέγχουμε αρχικά τις ειδικές περιπτώσεις εσωτερικών built-in εντολών exit, hist και cd και στην συνέχεια ο βασικός διαχωρισμός του κώδικα γίνεται με βάση την ύπαρξη piping ή όχι. Εάν δεν υπάρχει, τότε προχωρούμε στην δημιουργία μιας θυγατρικής διεργασίας και στην εκτέλεση της αντίστοιχης εντολής ελέγχοντας πρώτα για redirection, όπου επηρεάζουμε την είσοδο ή έξοδο της εκτέλεσης και για background case όπου η πατρική διεργασία (το κέλυφος) δεν περιμένει τον τερματισμό της θυγατρικής.

Όταν υπάρχει `ripping` η θυγατρική διεργασία δημιουργείται σε ξεχωριστό κομμάτι κώδικα όπου καλούμε μια συγκεκριμένη συνάρτηση για την εκτέλεση όλων των εντολών που αναγνωρίστηκαν στη σύνταξη του `ripping` και εκτελούνται ταυτόχρονα αντίστοιχοι έλεγχοι για `background case`.

Τέλος στο κέλυφος μας έχουμε προσθέσει την λειτουργία του ιστορικού. Εάν η είσοδος είναι έγκυρη, προστίθεται σε μια κυκλική λίστα για μελλοντική χρήση. Εκτελώντας την εντολή `hist`, εκτυπώνεται στην οθόνη το ιστορικό με ένα αναγνωριστικό από 1 έως 9. Καλώντας `hist #` όπου `#` το αναγνωριστικό, εκτελείται η αντίστοιχη εντολή απο το ιστορικό.

Κάποιες λεπτομέρειες που πρέπει να σημειωθούν για το κέλυφος είναι οι εξής:

1. Δεν υποστηρίζεται `redirection` με `ripping`.
2. Η συνάρτηση `executePipe` που εκτελεί τον κώδικα για το `ripping`, υποστηρίζει πολλαπλό `ripping` το οποίο εκτελείται εντός της συνάρτησης.
3. Στις περιπτώσεις λάθους όπως π.χ όταν υπάρχουν πολλά ορίσματα ή η είσοδος του χρήστη είναι μεγαλύτερη απο την αποδεκτή, ο χρήστης ενημερώνεται και η επανάληψη αρχίζει ξανά.
4. Η λειτουργία του `wildcard` υποστηρίζεται μόνο εφόσον το ειδικό σύμβολο `*` βρίσκεται στην αρχή της συμβολοσειράς εισόδου του χρήστη (πχ `gedit *g.txt`), αλλά όχι στο πρώτο όρισμα που αναπαριστά την εντολή.
5. Στην περίπτωση όπου η διεργασία είναι `background`, ο χρήστης πρέπει να τερματίσει την είσοδό του αμέσως μετά τον ειδικό χαρακτήρα `'&'`. π.χ για είσοδο : `mycommand &` (“λευκός χαρακτήρας”) δεν θα γίνει δεκτή απο το κέλυφός μας.
6. Στην περίπτωση του `redirection` μεταξύ του χαρακτηριστικού συμβόλου και του αρχείου, πρέπει να υπάρχει ακριβώς ένα κένο.

Ο κώδικας είναι πλήρως σχολιασμένος για την κατανόηση της ροής και του σκεπτικού ανάπτυξης.