

Αναφορά

myShell8804

Εμμανουήλ Χρήστος 8804

Ένα κέλυφος βοηθάει τον χρήστη να συνεργαστεί με το λειτουργικό σύστημα του υπολογιστή. Το κέλυφος που υλοποιείται χρησιμοποιεί διεπαφή γραμμής εντολών (CLI - command-line interface) και μπορεί να λειτουργήσει με δύο τρόπους, interactive και batch.

Αποτελείται από πλήθος συναρτήσεων, εκ των οποίων οι βασικότερες εκτελούνται σε βρόχο για να εξυπηρετούνται και οι δύο τρόποι λειτουργίας.

Παρακάτω περιγράφονται τα βασικά στοιχεία των συναρτήσεων για να κατανοηθεί η δομή και τα βασικά μέρη του συγκεκριμένου shell. Λεπτομέρειες του κώδικα περιγράφονται μόνο σε συγκεκριμένα, κρίσιμα για την κατανόηση σημεία.

Επιλογή Τρόπου Λειτουργίας

Η κλήση του προγράμματος από το τερματικό καθορίζει και τον τρόπο λειτουργίας του shell. Έτσι η πρώτη συνάρτηση που τρέχει στην main είναι η **FILE* desideIn(int argc, char **argv)**. Η συνάρτηση επιλέγει αν η είσοδος του προγράμματος θα είναι το πληκτρολόγιο (interactive λειτουργία) ή κάποιο αρχείο εντολών (batch λειτουργία) ανάλογα με το πλήθος των ορισμάτων που δόθηκαν στην κλήση του προγράμματος.

Εμφάνιση Prompt

Στην συνέχεια και μετά από κάποιες δηλώσεις μεταβλητών ξεκινάει ο κύριος βρόχος. Σε περίπτωση interactive λειτουργίας εκτελείται η **void printPrompt()**, η οποία τυπώνει το επώνυμο και το AEM (emmanouil_8804). Στην batch λειτουργία αυτό το βήμα παραλείπεται.

Διάβασμα Γραμμής

Προχωρώντας μέσα στην δομή επανάληψης εκτελείται η συνάρτηση **void readLine(FILE **input, char **line)**. Αποτελεί μία από τις κυριότερες συναρτήσεις του κελύφους καθώς είναι υπεύθυνη να διαβάζει μία γραμμή, από την είσοδο που επιλέχτηκε, και να την αποθηκεύει σε έναν πίνακα χαρακτήρων (string), μόνο όταν εξυπηρετούνται οι προϋποθέσεις που αφορούν το επιθυμητό μέγεθος (αναφέρονται στο README). Επίσης έχει ευθύνη για τον τερματισμό του κελύφους όταν το αρχείο εντολών που δύνεται σε batch λειτουργία δεν περιέχει εντολή εξόδου (η build-in εντολή εξόδου αναφέρετε παρακάτω).

Τεμαχισμός Γραμμής (Parsing)

Αφού ληφθεί η γραμμή, πρέπει να διατηρηθούν μόνο οι σημαντικοί χαρακτήρες και να τοποθετηθούν στην σωστή σειρά. Αυτό επιτυγχάνεται με την συνάρτηση **void parseLine(char *line, char ***tokens, int *lastTokenNum)**. Για να γίνει κατανοητή η λειτουργία της δύνεται ένα παράδειγμα, έστω ότι η γραμμή μας είναι η `ls -l`, τότε πρέπει να

γίνει απαλοιφή του κενού χαρακτήρα και τα string ls, -l να αποθηκευτούν σε συνεχόμενες θέσεις ενός πίνακα string (δισδιάστατο πίνακα χαρακτήρων). Αξίζει να αναφερθεί ότι η τελευταία θέση του πίνακα string πρέπει να είναι ο δείκτης NULL. Δηλαδή στο παράδειγμα αν τα ls, -l καταλαμβάνουν τις θέσεις 0, 1 αντίστοιχα, τότε η θέση 2 πρέπει να είναι ο δείκτης NULL.

Μορφή Εντολής

void execLine(char **tokens, int lastTokenNum), πρόκειται για την τελευταία βασική συνάρτηση που εκτελείται εντός του κύριου βρόχου της main. Το shell πρέπει να είναι σε θέση να εκτελεί πολλαπλές εντολές σε μία γραμμή. Για να γίνει κάτι τέτοιο, πρέπει ο χρήστης να διαχωρίζει τις εντολές με ειδικά σύμβολα όπως ; και &&. Κάθε σύμβολο αντιστοιχεί και σε έναν διαφορετικό τύπο εκτέλεσης των εντολών (αναφέρονται παρακάτω). Οπότε αυτό που κάνει η execLine είναι να αναζητά στον πίνακα των string την ύπαρξη κάποιο συμβόλου ή μη, και στην συνέχεια να προβαίνει στην κατάλληλη εκτέλεση. Σε περίπτωση ύπαρξης ειδικού χαρακτήρα διαχώρισης εντολών (EXDE), η συνάρτηση απομονώνει την πρώτη εντολή από τις επόμενες και προσέχει για την προϋπόθεση του NULL, που αναφερθήκαμε παραπάνω.

Μορφές Εκτέλεσης

Στο shell υπάρχουν τρεις μορφές εντολής:

Απλή Εντολή : δεν βρέθηκε EXDE => κύρια εκτέλεση

Εντολή Ερωτηματικού (;) : **void execQues(char ***commands, int delPos, int restlastTokenNum)**, οδηγεί την απομονωμένη εντολή στην κύρια εκτέλεση και στην συνέχεια καλεί την execLine για τον υπόλοιπο πίνακα των string.

Εντολή Διπλού Και (&&) : **void execAmber(char ***commands, int delPos, int restlastTokenNum)**, οδηγεί την απομονωμένη εντολή στην κύρια εκτέλεση και στην συνέχεια καλεί την execLine για τον υπόλοιπο πίνακα των string ΜΟΝΟ αν η προηγούμενη εντολή εκτελέστηκε επιτυχμένα.

Κύρια Εκτέλεση

Η **int execSimple(char **tokens, int lastTokenNum)** είναι υπεύθυνη για την διαχείριση απλών εντολών. Σε περίπτωση κενής γραμμής επιστρέφει στην αρχή του βρόχου και στην περίπτωση της εντολής quit, που αποτελεί την build-in εντολή τερματισμού του shell, εγκαταλείπει το κέλυφος. Σε διαφορετική περίπτωση με την βοήθεια της fork(), δημιουργεί έναν απόγονο ώστε να μπορούν να τρέξουν εντολές Linux, χωρίς να εγκαταλείπεται το βασικό πρόγραμμα.

Ο απόγονος ελέγχει άμα στο κομμάτι της απλής εντολής υπάρχουν σύμβολα ανακατεύθυνσης εισόδου-εξόδου (<, >). Αν βρεθεί σύμβολο υλοποιεί την αντίστοιχη ανακατεύθυνση με την βοήθεια της dup2(). Και τέλος εκτελεί την εντολή Linux με την execvp(). Ο απόγονος θα τερματιστεί μέσω της execvp() ή με μιας εντολή exit() σε περίπτωση αποτυχίας της execvp().

Ο πατέρας αποτελεί την διεργασία που τρέχει το βασικό πρόγραμμα και έχει τεθεί σε

αναμονή (wait(&status)) για την εκτέλεση του απογόνου του. Μετά τον τερματισμό του απογόνου ο πατέρας επιστρέφει τιμή 0 ή 1 για αποτυχημένη ή επιτυχημένη εκτέλεση αντίστοιχα (έλεγχος του status).

Build-in Εντολή Εξόδου

Η **void quitShell()** εκτελείται όταν δοθεί η εντολή quit, τυπώνει μήνυμα στην οθόνη και εγκαταλείπει το κέλυφος.

Σε όλο το μήκος του προγράμματος υπάρχουν συνθήκες έλεγχου. Μερικές οδηγούν το κέλυφος στον τερματισμό ενώ μερικές οδηγούν στην αρχή του βρόχου ζητώντας την επόμενη γραμμή προς εκτέλεση.

Τέλος παραθέτονται στιγμιότυπα του κελύφους σε διάφορες περιπτώσεις :

```
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$ ./myShell8804
Welcome to myShell8804 created by : Emmanouil Christos 8804

emmanouil_8804>ls
cmd Makefile myShell8804 myShell8804.c
emmanouil_8804>ls -l > example
emmanouil_8804>cat < example
total 36
-rw-r--r-- 1 blkderivation blkderivation 84 Iav 17 18:01 cmd
-rwx----- 1 blkderivation blkderivation 0 Iav 17 20:28 example
-rw----- 1 blkderivation blkderivation 170 Iav 15 05:14 Makefile
-rwxr-xr-x 1 blkderivation blkderivation 17472 Iav 17 17:52 myShell8804
-rwxrwx-rw- 1 blkderivation blkderivation 7300 Iav 17 15:28 myShell8804.c
emmanouil_8804>cat < falsefilename
Error : open() Failure 1
emmanouil_8804>ls ; pwd && falsecommand
cmd example Makefile myShell8804 myShell8804.c
/home/blkderivation/Desktop/teliko
ERROR : execvp() Failure
emmanouil_8804>ls ; falsecommans && pwd
cmd example Makefile myShell8804 myShell8804.c
ERROR : execvp() Failure
emmanouil_8804>> example pwd
emmanouil_8804>< example cat
/home/blkderivation/Desktop/teliko
emmanouil_8804>ls > example1 ; > example1 pwd && cat < example1 && quit
/home/blkderivation/Desktop/teliko
GoodBye!!!
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$
```

```
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$ ./myShell8804 cmd
Welcome to myShell8804 created by : Emmanouil Christos 8804

batch Function with quit command
file text:
echo batch Function with quit command
echo file text:
cat cmd
echo file text end
quit
file text end
GoodBye!!!
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$
```

```
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$ ./myShell8804 cmd
Welcome to myShell8804 created by : Emmanouil Christos 8804

batch Function without quit command
file text:
echo batch Function without quit command
echo file text:
cat cmd
echo file text end
file text end
EOF or ERROR : fgets() Failure
blkderivation@blkderivation-virtual-machine:~/Desktop/teliko$
```

Περιεχόμενο αρχείου README.md

```
1 # myShell8804
2 To myShell8804 είναι ένας διερμηνέας γραμμής εντολών, ικανός να τρέξει σε interactive ή batch λειτουργία.
3
4 ##Author
5 Εμμανουήλ Χρήστο AEM:8804
6
7 ## Run
8 Εγκαταστήστε το make utility στα linux:
9 ```sh
10 $ sudo apt install make
11 ```
12
13 Ανοίξτε το τερματικό στον ίδιο φάκελο με τα αρχεία Makefile και myShell8804.c,
14 και στην συνέχεια πληκτρολογήστε:
15 ```sh
16 $ make
17 ```
18
19 Εκκίνηση σε Interactive Λειτουργία:
20 ```sh
21 $ ./myShell8804
22 ```
23
24 Η εκκίνηση σε Batch Λειτουργία:
25 ```sh
26 $ ./myShell8804 filename
27 ```
28
29 ## Features
30 Μέγεθος 1024 χαρακτήρες
31 Μέγιστο μέγεθος γραμμής 512 χαρακτήρες
32
33 Οι εντολές που δίνονται από τον χρήστη μπορεί να είναι:
34
35 ##### Εντολές των Linux
36 Κάθε εντολή που τρέχει στο terminal και ΔΕΝ αλλάζει το directory είναι επιτρεπτή.
37
38 ##### Build-In Εντολές
39 quit : Τυπώνει GoodBye!!! και πραγματοποιείτε έξοδο από το κέλυφος.
40
41 Το συγκεκριμένο shell επιτρέπει την ύπαρξη περισσότερων εντολών ανά γραμμή:
42 - ; : οι εντολές που διαχωρίζονται με τον χαρακτήρα ; θα πρέπει να εκτελούνται διαδοχικά, ανεξάρτητα από την επιτυχία ή μη ολοκλήρωση της πρώτης εντολής.
43 - && : οι εντολές που διαχωρίζονται με τον χαρακτήρα && θα πρέπει να εκτελούνται οι διαδοχικές εντολές εφόσον η προηγούμενη επιστρέψει επιτυχώς.
44
45 Επιπλέον κάθε εντολή υποστηρίζει redirections:
46 - < : όταν αυτό το σύμβολο ακολουθείτε από κάποιο όνομα υπαρκτού αρχείου ανακατευθύνει την είσοδο της εντολής.
47 - > : όταν αυτό το σύμβολο ακολουθείτε από κάποιο όνομα αρχείου ανακατευθύνει την έξοδο της εντολής (αν δεν υπάρχει το αρχείο το δημιουργεί με πλήρη δικαιώματα).
48
49 Το shell ΔΕΝ υποστηρίζει pipes!!!
50
51 Μεταξύ εντολών, ορισμάτων και ειδικών χαρακτήρων πρέπει να υπάρχουν κενά!!!
52
53 ## Errors
54 Τα σφάλματα που προκύπτουν εμφανίζουν κατάλληλο μήνυμα. Μετά απο κάποιο σφάλμα ίσως υπάρξει τερματισμός του κελύφους.
```