

ΕΡΓΑΣΙΑ

ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

Έτος: 2016-2017

Περιγραφή

Ζητείται να υλοποιήσετε ένα διερμηνέα γραμμής εντολών (**command line interpreter**) ή κέλυφος (**shell**). Το κέλυφος θα πρέπει να μπορεί να λειτουργήσει με δύο τρόπους, **interactive** και **batch**.

Interactive λειτουργία

Στην **interactive** λειτουργία, το κέλυφος σας θα ξεκινά εκτελώντας το όνομα που θα αντιστοιχεί στο εκτελέσιμο αρχείο σας, πχ

```
$ ./myshell
```

Με την εκτέλεση του προγράμματός σας θα εμφανίζεται ένα prompt (μια συμβολοσειρά με το επίθετό σας και το αριθμό μητρώου σας πχ, `kalogiannis_3695>`) και ο χρήστης του shell θα πληκτρολογεί μια εντολή στη γραμμή, πχ

```
kalogiannis_3695> ls
```

Με την πληκτρολόγηση της εντολής (στο prompt του), το shell θα δημιουργεί μια διαδικασία παιδί (**child process**) που θα εκτελεί την εντολή που έχετε εισάγει και στη συνέχεια εφόσον ολοκληρωθεί η εκτέλεσή της να επιστρέφει στο prompt. Το shell που θα υλοποιήσετε, θα πρέπει να λαμβάνει φροντίδα να υποστηρίζει παραμέτρους εντολών πχ

```
kalogiannis_3695> ls -l
```

αλλά και τη δυνατότητα να δίνονται περισσότερες από μια εντολές διαχωρισμένες με το χαρακτήρα `;`, πχ

```
kalogiannis_3695> ls -l ; cat file
```

Στη περίπτωση αυτή οι εντολές που διαχωρίζονται με τον χαρακτήρα `;` θα πρέπει να εκτελούνται ταυτόχρονα (το πλήθος των **child process** που θα δημιουργηθούν θα πρέπει να ισοδυναμεί με το πλήθος των εντολών που διαχωρίζονται με τον χαρακτήρα `;`). Το shell σας δε θα πρέπει να τυπώνει το prompt του (πχ `kalogiannis_3695>`), πριν ολοκληρωθεί η εκτέλεση όλων

των διεργασιών (η χρήση των εντολών `wait()` και/η `waitpid()` θα σας φανεί χρήσιμη στο σημείο αυτό).

Για παράδειγμα η εκτέλεση των εντολών

```
kalogiannis_3695> ls -l ; cat file ; grep hello file2
```

θα πρέπει να γίνεται ταυτόχρονα ώστε οι απαντήσεις των εντολών να εμφανίζονται στην οθόνη με τυχαίο τρόπο με τον οποίο εκτελούνται οι εντολές από τον χρονοδρομολογητή του λειτουργικού συστήματος.

Για τη έξοδο από το shell θα πρέπει να πληκτρολογήσετε την λέξη **quit** (η χρήση της κλήσης συστήματος `exit()` θα σας φανεί χρήσιμη στο σημείο αυτό). Αν η λέξη **quit** υπάρχει σε γραμμή με άλλες εντολές θα πρέπει να διασφαλίσετε ότι οι υπόλοιπες εντολές θα εκτελεστούν και θα ολοκληρωθούν προτού εγκαταλείψετε το πρόγραμμα σας, πχ

```
kalogiannis_3695> quit ; cat file
```

ή

```
kalogiannis_3695> cat file ; quit
```

Και στις δυο παραπάνω περιπτώσεις θα εκτελεστεί και θα ολοκληρωθεί η `cat file` και μετά θα γίνει έξοδος από το πρόγραμμα σας.

Batch λειτουργία

Σε **batch** λειτουργία, το κέλυφος σας θα ξεκινά καθορίζοντας ένα αρχείο δέσμης (batchfile) στη γραμμή εντολών του, πχ

```
$ ./myshell batchfile
```

Το **batchfile** θα είναι ένα αρχείο κειμένου που θα περιλαμβάνει το σύνολο των εντολών που θα πρέπει να εκτελεστούν από το **shell** σας, πχ

```
ls -l
pwd
ps
touch hello
ls -l ; cat file ; grep foo file2
quit ; cat hello
```

Το **batchfile** θα μπορεί να περιλαμβάνει είτε μια εντολή ανά γραμμή, είτε περισσότερες εντολές ανά γραμμή (με διαχωριστικό χαρακτήρα το ;). **Σε κάθε περίπτωση η λογική της εκτέλεσης των εντολών (mother-child) θα είναι ίδια με αυτή της περίπτωσης του Interactive mode.**

Προδιαγραφές προγράμματος

Το πρόγραμμά σας θα πρέπει να υποστηρίζει και τις δυο μορφές λειτουργίας του, batch και interactive. Σε περίπτωση λαθεμένης εισαγωγής εντολών ή λάθους μορφής batchfile θα πρέπει να εμφανίζεται μήνυμα λάθους στον χρήστη διακόπτοντας τη εκτέλεση του προγράμματος σας (για παράδειγμα αν η εντολή που ζητάτε προς εκτέλεση δεν υπάρχει, έχει λάθος σύνταξης ή το batchfile είναι άδειο).

Αναφορικά με το πλήθος των εντολών που μπορούν να δοθούν σε interactive mode ανά γραμμή θα πρέπει να φροντίσετε το μέγεθος να μην υπερβαίνει τους 512 χαρακτήρες συμπεριλαμβανομένων και των κενών χαρακτήρων καθώς και των διαχωριστικών χαρακτήρων (;). Σε batchfile mode κάθε γραμμή θα πρέπει να έχει το μέγιστο 512 χαρακτήρες συμπεριλαμβανομένων και των κενών χαρακτήρων καθώς και των διαχωριστικών χαρακτήρων (;).

Επιπρόσθετα θα πρέπει να φροντίσετε να διαχειριστείτε τυχαίες καταστάσεις που δεν είναι λανθασμένες όπως:

- Μια άδεια γραμμή σε batchfile ή άδεια εντολή σε interactive mode,
- Επιπλέον κενοί χαρακτήρες μέσα στο batchfile ή στο interactive mode.

Interactive mode	Batch mode (μορφή batchfile)
kalogiannis_3695>ls -l; ;ls	cat
	ls -l;
	echo "hello" ; ;ls -a

- Batchfile που δεν τελειώνει με την εντολή quit() ή δεν την περιλαμβάνει.

Χρήσιμες πληροφορίες

Το κέλυφός σας στην ουσία θα είναι ένα loop το οποίο θα τυπώνει επαναληπτικά το prompt σας (στην περίπτωση του Interactive mode), θα κάνει «parsing» το «input» σας, θα χωρίζει το Input

σας σε τμήματα (πλήθος εντολών), θα κάνει τους αναγκαίους ελέγχους περί ορθότητας των εντολών σας και θα τις εκτελεί με τη προαναφερθείσα λογική.

Parsing

Για εισαγωγή-ανάγνωση των γραμμών μπορείτε να χρησιμοποιήσετε την μέθοδο **fgets()**. Για να ανοίξετε ένα αρχείο δείτε την μέθοδο **fopen()**. Η μέθοδος **strtok()** θα σας φανεί ιδιαίτερα χρήσιμη καθώς διαβάζει μια ολόκληρη γραμμή και «τεμαχίζει» τη γραμμή σε ανάλογα τμήματα.

Εκτέλεση εντολών

Στα πλαίσια της εργασίας κάντε χρήση μόνο της εντολής `execvp()`. Θυμηθείτε ότι επιτυχής εκτέλεση της εντολής δεν επιστρέφει κάτι, διαφορετικά (σε περίπτωση λάθους, πχ η εντολή που ζητήθηκε να εκτελεσθεί δεν υπάρχει) επιστρέφει -1. Το πιο ενδιαφέρον στο σημείο αυτό είναι η τοποθέτηση των ορισμάτων με τον σωστό τρόπο:

```
int execvp(const char *file, char *const argv[]);
```

Το πρώτο όρισμα αφορά το πρόγραμμα-εντολή που πρόκειται να εκτελεσθεί μαζί με την πλήρη διαδρομή του(path). Το δεύτερο όρισμα, μπορεί να προέρχεται από την ίδια τη συνάρτηση «main» του προγράμματος σας :

```
int main(int argc, char *argv[]);
```

Παραδοτέο υλικό

Για την εργασία αυτή θα πρέπει να παραδώσετε :

- Τον πηγαίο κώδικα σας (όχι object files ή executables)
 - Ένα Makefile για τη μεταγλώττιση του κώδικα σας
 - Ένα αρχείο README με ορισμένες βασικές πληροφορίες τεκμηρίωσης του κώδικα σας
-