

Άσκηση 1

Λειτουργικά Συστήματα (HY345)

1. Περιγραφή

Σε ένα λειτουργικό σύστημα, το κέλυφος είναι η κύρια διεπαφή που επιτρέπει στον χρήστη να αλληλοεπιδράσει με τον υπολογιστή, παρέχοντας του μία γραμμή εντολών ή ένα γραφικό περιβάλλον για να εκτελέσει εντολές, να διαχειριστεί αρχεία και να εκτελέσει προγράμματα. Δρα ως ένας διαμεσολαβητής μεταξύ του χρήστη και του πυρήνα του λειτουργικού συστήματος, διερμηνεύοντας την είσοδο του σε εντολές που ο πυρήνας μπορεί να καταλάβει και να εκτελέσει. Κοινά παραδείγματα κελυφών είναι το `bash` και το `Z shell (zsh)` σε Linux και macOS και η γραμμή εντολών στα Windows.

Σε αυτή την άσκηση θα υλοποιήσετε ένα κέλυφος χρησιμοποιώντας τη γλώσσα C και θα εξοικειωθείτε με τον τρόπο λειτουργίας του. Πιο συγκεκριμένα, θα μάθετε τον τρόπο δημιουργίας και ελέγχου διεργασιών, καθώς επίσης το τι είναι οι σωληνώσεις και πως υλοποιούνται. Τέλος, θα δείτε πως γίνεται η ανακατεύθυνση εισόδου-εξόδου σε εντολές.

2. Υλοποίηση

2.1 Ονομασία

Το κέλυφος θα ονομάζεται `hy345sh` και στο τερματικό το σύμβολο που εμφανίζεται θα πρέπει να είναι της μορφής `<Χρήστης>@<AM>-hy345sh:< Κατάλογος >$`, όπου:

- `<AM>`: Ο αριθμός μητρώου σας. π.χ. `csd5072`
- `<Χρήστης>`: Ο χρήστης που είναι συνδεδεμένος στο σύστημα.
- `<Κατάλογος>`: Ο τρέχον κατάλογος εργασίας.

2.2 Απλές Εντολές

Το κέλυφος θα πρέπει να μπορεί να διαβάζει και να εκτελεί εντολές από το χρήστη. Για κάθε εντολή θα δημιουργεί μια διεργασία παιδί, η οποία με τη σειρά της θα εκτελεί την εντολή του χρήστη. Θα πρέπει να υποστηρίζονται και πολλές εντολές μαζί διαχωρισμένες με `;"` καθώς και εντολές που έχουν ορίσματα. Μπορείτε να χρησιμοποιήσετε εντολές όπως `ls, date, cat y1; rm y1, mkdir test, cd` κ.α για να ελέγξετε τη σωστή λειτουργία του κελύφους σας.

2.3 Παγκόσμιες Μεταβλητές

Από το κελυφός σας θα πρέπει να μπορεί ο χρήστης να ορίζει και να διαβάζει παγκόσμιες μεταβλητές. Για παράδειγμα, χρησιμοποιώντας την εντολή `x=world` θα δημιουργείται η παγκόσμια μεταβλητή με την τιμή `world`. Αργότερα χρησιμοποιώντας την εντολή `echo hello $x` θα πρέπει να τυπώνεται στην κονσόλα `hello world`

2.4 Σωληνώσεις

Οι σωληνώσεις μας δίνουν την δυνατότητα να διοχετεύουμε την τυπική έξοδο μιας εντολής απευθείας στην τυπική είσοδο μιας άλλης. Το κέλυφος σας θα πρέπει να υποστηρίζει τέτοιες σωληνώσεις με τη χρήση του συμβόλου "|".

2.5 Ανακατεύθυνση

Οι περισσότερες διεργασίες που ξεκινούν από εντολές γράφουν στην τυπική έξοδο-**stdout** (δηλαδή, γράφουν στην οθόνη του τερματικού) και πολλές παίρνουν την είσοδό τους από την τυπική είσοδο-**stdin** (δηλαδή τη διαβάζουν από το πληκτρολόγιο). Υπάρχει επίσης το τυπικό σφάλμα-**stderr**, όπου οι διεργασίες γράφουν τα μηνύματα σφάλματός τους, από προεπιλογή, στην οθόνη του τερματικού.

Με την ανακατεύθυνση μπορούμε να αλλάξουμε την τυπική είσοδο και την τυπική έξοδο εντολών. Το κέλυφος σας θα πρέπει να υποστηρίζει:

- **Ανακατεύθυνση εισόδου:**

Χρησιμοποιώντας το σύμβολο "<" θα πρέπει τα περιεχόμενα του αρχείου που ακολουθεί να περνούν σαν είσοδος στην εντολή. π.χ. `cat < test.txt`

- **Ανακατεύθυνση εξόδου:**

Χρησιμοποιώντας το σύμβολο ">" θα πρέπει η τυπική έξοδος της εντολής να ανακατευθύνεται στο αρχείο που ακολουθεί. Τυχόν προηγούμενα περιεχόμενα του αρχείου θα πρέπει να διαγράφονται. π.χ `ls -al > x.txt` - Η τυπική έξοδος της εντολής `ls -al` τυπώνεται στο αρχείο `x.txt`.

Επιπλέον, με το σύμβολο ">>" θα πρέπει η έξοδος της εντολής να ανακατευθύνεται στο αρχείο που ακολουθεί, αλλά εδώ θα πρέπει η έξοδος να προστίθεται στο τέλος αυτού, χωρίς να υπάρχει διαγραφή τυχόν προηγούμενων περιεχομένων. π.χ. `ls >> y.txt` - Τα ονόματα των αρχείων που τυπώνει η εντολή `ls` προστίθενται στο τέλος του αρχείου `y.txt`

2.6 Έλεγχος ροής προγράμματος και επαναληπτικοί βρόγχοι

Το shell σας θα πρέπει να υποστηρίζει έλεγχο ροής (**if statements**) και επαναληπτικούς βρόγχους (**for loops**). Συγκεκριμένα, το πρόγραμμα που θα υλοποιήσετε θα πρέπει να:

1. Να αναγνωρίζει και να ερμηνεύει τη δομή υπό συνθήκη εκτέλεσης:

```
if [ condition ];
```

```
then
```

```
<εντολές>
```

```
fi
```

Όπου η συνθήκη μπορεί να είναι μια απλή αριθμητική/λογική έκφραση ή μια τιμή μεταβλητής. **Προσοχή**, για να γράψετε σωστά το condition θα πρέπει να το βάλετε σε αγκύλες με κενό, και όχι κολλητά.

2. Να υποστηρίζει την επαναληπτική δομή τύπου for loop:

for x in <λίστα στοιχείων>;

do

<εντολές>

done

Όπου η λίστα στοιχείων μπορεί να αποτελείται από αριθμούς ή strings, τα οποία θα αποδίδονται στη μεταβλητή x σε κάθε επανάληψη.

3. Εκτέλεση

Σε περίπτωση που ένα πρόγραμμα μπορεί να οδηγήσει σε κατάρρευση του συστήματος μπορούμε, αντί να το τρέξουμε στο σύστημα μας ή σε ένα κοινόχρηστο σύστημα, να το τρέξουμε σε μια εικονική μηχανή ή σε έναν εξομοιωτή. Έτσι, σε περίπτωση κατάρρευσης θα επηρεαστεί μόνο η εικονική μηχανή/εξομοιωτής, που μπορεί πολύ εύκολα να επανεκκινηθεί/επιδιορθωθεί.

3.1 Εικονικός Δίσκος

Σε αυτή την άσκηση θα πρέπει να τρέξετε το πρόγραμμα σας χρησιμοποιώντας τον εξομοιωτή QEMU. Ο QEMU είναι ήδη εγκατεστημένος στα μηχανήματα της σχολής. Αυτός διαβάζει έναν εικονικό δίσκο, ο οποίος μπορεί να περιέχει ένα οποιοδήποτε λειτουργικό σύστημα. Για την συγκεκριμένη άσκηση, υπάρχει ένας έτοιμος εικονικός δίσκος με Linux OS στην περιοχή του μαθήματος (~hy345/qemu-linux/hy345-development.img). Εσείς, θα πρέπει να τον αντιγράψετε στην περιοχή σας πρώτου τον χρησιμοποιήσετε. Αυτό μπορείτε να το κάνετε με την εντολή

cp ~hy345/qemu-linux/hy345-development.img .

αυτή η εντολή θα αντιγράψει τον εικονικό δίσκο στο directory που είστε εκείνη τη στιγμή. Προσοχή η τελεία στο τέλος είναι σημαντική.

Στην περίπτωση που κατά την αντιγραφή, σας εμφανιστεί το μήνυμα "Disk quota exceeded" (δεν υπάρχει αρκετός χώρος για αντιγραφή) ενδέχεται, ενώ σας εμφανίζεται ο εικονικός δίσκος, αυτός να μην δουλεύει σωστά. Για να μη χρειαστεί να διαγράψετε τα αρχεία σας από την περιοχή σας, μπορείτε να χρησιμοποιήσετε το /spare (Κάντε cd /spare για να πάτε εκεί). Φτιάξτε ένα φάκελο εκεί με τον Αριθμό Μητρώου σας και κλειδώστε τον, όπως είχατε κλειδώσει και την περιοχή σας. Έπειτα μπορείτε να αντιγράψετε εκεί τον εικονικό δίσκο. Προσοχή, ο φάκελος αυτός υπάρχει μόνο στο μηχανήμα που τον δημιουργήσατε (π.χ. despolo), άρα για να τον χρησιμοποιείτε θα πρέπει να συνδέεστε εκεί.

3.2 Είσοδος στον QEMU

Εφόσον, έχετε τον εικονικό δίσκο στην περιοχή σας μπορείτε να τρέξετε τον QEMU χρησιμοποιώντας την εντολή

qemu-system-i386 -hda <path to virtual disk image> -display curses

όπου με το όρισμα -hda <path to virtual disk image> υποδεικνύουμε στον QEMU που θα βρει τον εικονικό δίσκο και με το όρισμα **-display curses** ενεργοποιούμε τη λειτουργία χωρίς γραφικό περιβάλλον (μόνο γραμμή εντολών).

Αφού τρέξετε την παραπάνω εντολή θα ξεκινήσει το λειτουργικό σύστημα και θα σας ζητηθεί το όνομα χρήστη και ο κωδικός. Αυτά είναι και τα δύο "hy345".

3.3 Μεταφορά αρχείων

Για να μεταφέρετε αρχεία από το μηχάνημα της σχολής, στο φιλοξενούμενο λειτουργικό σύστημα που τρέχει στον QEMU, μπορείτε μέσα στο φιλοξενούμενο λειτουργικό σύστημα να τρέξετε την εντολή `scp` με την εικονική διεύθυνση διαδικτυακού πρωτοκόλλου 10.0.2.2. Για παράδειγμα, για να μεταφέρετε το αρχείο "test.txt" από το μηχάνημα της σχολής στο λειτουργικό που τρέχει στον QEMU μπορείτε να χρησιμοποιήσετε την εντολή:

```
scp <username>@10.0.2.2:~/test.txt .
```

Ενώ για το αντίθετο μπορείτε να χρησιμοποιήσετε την εντολή:

```
scp test.txt <username>@10.0.2.2: ~
```

όπου <username> το όνομα χρήστη σας. Για να τρέξουν οι παραπάνω εντολές ίσως σας ζητηθεί να εισάγετε τον κωδικό σας για τα μηχανήματα της σχολής.

3.4 Αποθήκευση αρχείων

Σε περίπτωση που το λειτουργικό σύστημα που τρέχει στο QEMU καταρεύσει απροσδόκητα, υπάρχει περίπτωση να χάσετε αρχεία που τυχόν έχετε σε αυτό. Για να το αποτρέψετε, σας παρέχεται η εντολή `save`, η οποία προτείνεται να καλείται πριν την εκτέλεση του προγράμματος σας. Αυτή μειώνει τις πιθανότητες να χάσετε τα αρχεία σας. Σε κάθε περίπτωση, είναι πολύ σημαντικό, να κρατάτε τα αρχεία του προγράμματος σας και εκτός του εξομοιωτή.

3.5 Έξοδος από QEMU

Για να αποχωρήσετε από τον QEMU αρκεί να εκτελέσετε την εντολή `turnoff`. Αυτή εξασφαλίζει μια ομαλή απενεργοποίηση του λειτουργικού συστήματος και μειώνει την πιθανότητα να χάσετε τα αρχεία σας.

3.6 Απομακρυσμένος Έλεγχος

- **Windows**

Για να συνδεθείτε με τα μηχανήματα της σχολής και να μεταφέρετε τα αρχεία σας μπορείτε να χρησιμοποιήσετε το WinSCP ή κάποιο άλλο παρόμοιο πρόγραμμα, ενώ για τον απομακρυσμένο έλεγχο μπορείτε να χρησιμοποιήσετε κάποιο πρόγραμμα `secure ssh`, όπως το **putty**. Αναλυτικές οδηγίες μπορείτε να βρείτε στη κάνοντας κλικ [εδώ](#)

- **Linux**

Αν δουλεύετε απομακρυσμένα σε κάποιο μηχάνημα του τμήματος, για να ξεκινήσετε τον QEMU στο απομακρυσμένο μηχάνημα θα πρέπει να συνδεθείτε με X11 forwarding από τον δικό σας υπολογιστή. Όταν κάνετε `enable` το μηχάνημα π.χ. `despolo` θα σας επιστρέψει την εντολή που χρειάζεται να εκτελέσετε για να συνδεθείτε. π.χ.:

```
ssh username@despolo.csd.uoc.gr -p 22 -Y
```

Δοκιμάστε να τρέξετε `xterm`, θα πρέπει να σας ανοίξει το `xterm`. Εφόσον λειτουργεί το `xterm`, μπορείτε να χρησιμοποιήσετε τον QEMU κανονικά. Εναλλακτικά, μπορείτε απλά να τρέχετε

τον QEMU χωρίς γραφικό περιβάλλον(πολύ λιγότερο lag) με την βιβλιοθήκη `-display curses` (προτεινόμενο):

`qemu-system-i386 -hda hy345-development.img -display curses`

4. Χρήσιμες Συμβουλές

Για να σιγουρευτείτε ότι το πρόγραμμα σας δουλεύει σωστά μπορείτε να δοκιμάσετε και περίπλοκες εντολές που περιέχουν και ανακατευθύνσεις και σωληνώσεις, καθώς και να συγκρίνετε το κέλυφός σας με αυτό στα μηχανήματα της σχολής. Παρακάτω παρατίθενται μερικές τέτοιες εντολές:

- `ls -al>y1`
- `cat y1;rm y1`
- `age=20; echo I am $age years old`
- `ls |sort|uniq|wc`
- `echo hello > yy; echo world >> yy; cat < yy | sort | uniq | wc > y1; cat y1`
- `func="5+5="; res=10; echo $func$res`
- `my_var="This is my var"; echo $my_var | wc > xx; cat<xx`
- `x="5|5;10"; echo $x>file; cat<file`
- `x=5; if [ $x == 5 ] ; then echo "hello world"; fi`
- `x=5`
`if [ $x == 5 ]`
`then`
`echo $x`
`fi`
- `for i in 1 2 3; do echo $i; done`
- `for l in "h" "e" "l" "l" "o"`
`do`
`echo $i`
`done`

Προσοχή, όπως παρατηρείτε στις παραπάνω εντολές δεν είναι ανάγκη να υπάρχουν κενά μεταξύ των διάφορων συμβόλων `","`, `"|"`, `">"`, καθώς και αν υπάρχουν, αυτά παραλείπονται.

5. Οδηγίες Παράδοσης

Η παράδοση της άσκησης θα γίνει μέσω **E-Learn** (ο σύνδεσμος του μαθήματος είναι [εδώ](#)).

Θα πρέπει να φτιάξετε ένα zip αρχείο το οποίο θα περιέχει:

- Τον κώδικα σας, ο οποίος θα περιέχει σχόλια που να τον εξηγούν.
- Ένα **Makefile** το οποίο με την χρήση των εντολών:
 - **make all** να παράγεται το εκτελέσιμο
 - **make clean** να διαγράφονται τα αρχεία που δημιουργήθηκαν.
- Ένα **README** στο οποίο θα εξηγείτε τη δουλειά σας, και ότι σημειώσεις θεωρείτε ότι είναι σημαντικές.

Μην βάλετε στο zip αρχείο σας τον εικονικό δίσκο.

6. Χρήσιμοι σύνδεσμοι

Παρακάτω παρατίθενται οι οδηγίες χρήσης κάποιων εντολών που ίσως χρειαστείτε:

- fork(2): <https://man7.org/linux/man-pages/man2/fork.2.html>
- exec(3): <https://man7.org/linux/man-pages/man3/exec.3.html>
- setenv(3): <https://www.man7.org/linux/man-pages/man3/setenv.3.html>
- getenv(3): <https://man7.org/linux/man-pages/man3/getenv.3.html>
- pipe(2): <https://man7.org/linux/man-pages/man2/pipe.2.html>
- dup2(2): <https://man7.org/linux/man-pages/man2/dup.2.html>
- termios(3): <https://man7.org/linux/man-pages/man3/termios.3.html>

7. Παρατηρήσεις

- Η άσκηση αυτή είναι καθαρά ατομική. Τα παραδοτέα αρχεία θα περαστούν απο σύστημα ανίχνευσης αντιγραφών. Τυχούσες αντιγραφές συνεπάγονται με αυτόματο μηδενισμό.
- Στην αρχή του κώδικά σας, συμπεριλάβετε σε σχόλια το ονοματεπώνυμο και τον αριθμό μητρώου σας.
- Κρατήστε περιποιημένο τον κώδικα σας και προσθέστε σχόλια που να εξηγούν τη λειτουργία του όπου χρειάζεται.
- Η χρήση κώδικα ο οποίος έχει παραχθεί από AI **απαγορεύεται ρητά**