Κ24: Προγραμματισμός Συστήματος - 1η Εργασία

ΑΝΤΡΕΪ-ΑΝΤΡΙΑΝ ΠΡΕΝΤΑ - 1115202000263

Το πρόγραμμα, με εξαίρεση την διαχείριση aliases και την εντολή cd, υποστηρίζει όλες τις εντολές που απαιτεί η εργασία. Η strsplt έχει leaks. Επίσης, η σωλήνωση δεν λειτουργεί για πολύ μεγάλες εξόδους (όπως η δοκιμαστική εντολή για το αρχείο gene_with_protein_product). Πιστεύω όμως ότι αυτό έχει να κάνει με το περιορισμένο μέγεθος των pipes.

Γενική δομή:

Η γενική δομή του κελύφους είναι:

- περιμένει για input (με σταθερό μέγεθος των 256 χαρακτήρων)

- δέχεται γραμμή εισόδου

- εκτελεί τις εντολές από την γραμμή εισόδου σειριακά (ή οχι αν έχουν δηλωθεί ως background)

- γυρνάει στην αρχή και περιμένει για input

Το κέλυφος αρχικά δέχεται μία γραμμή εισόδου, η οποία γραμμή αυτή μπορεί να περιέχει πολλές εντολές διαχωρισμένες με ‘;’. Ο διαχωρισμός αυτός γίνεται με την χρήση της strtok (η οποία χρησιμοποιείται σε πολλά κομμάτια του προγράμματος).

Πρωτού εκτελεστούν οι εντολές αυτές, ελέγχονται μήπως πρειέχουν κάποιο redirection (και τι είδος), αν είναι για background, αν περιέχουν σωλήνωση κλπ. Όλες αυτές οι πληροφορίες μεταφέρονται στην διεργασία παιδί, η οποία, με βάση των πληροφοριών αυτών, εκτελεί την εντολή με την κλήση της norm_exec. Σκοπός της διεργασίας παιδί είναι να αντικατασταθεί αυτή από την exec για να μην χαθεί η διεργασία γονέας.

Συναρτήσεις και διάφοες σχεδιαστικές επιλογές:

Ο διαχωρισμός της εντολής απαιτεί τη συνεχόμενη κλήση της strtok() για το ίδιο string. Αυτό είναι προβληματικό στην περίπτωσή μας, αφού η strtok() καλείται και για την γραμμή εισόδου (‘;’), αλλά και για τις εντολές μέσα στην γραμμή εισόδου (‘|’). Για αυτό χρησιμοποιείται η strsplt(), η οποία δουλεύει παρόμοια με την strtok(), μόνο που το υπόλοιπο του string αποθηκεύεται.

α) Ανακατευθύνσεις

Θεωρείτα ότι για κάθε εντολή, το πολύ μία ανακατεύθνση εξόδου έχουμε και όχι συνεχόμενες (όχι δηλαδή “cat in.txt > out.txt > out2.txt >…"). Η συνάρτηση redir_detect() χρησιμοποιείται για την ανίχνευση των ανακατευθύνσεων με μία απλή σειριακή αναζήτηση σε όλο το input. Το ‘<’ αντικαθίσταται με κενό (αφού cat in.txt είναι παρόμοιο με cat < in.txt). Ανάλογα με το redirection (> ή >>), το αρχείο ανοίγει και το αποτέλεσμα αποθηκεύεται ή προσαρτάται σε αυτό. Εδώ χρησιμοποιείται η strtok για να χωρίσουμε την εντολή από το όνομα του αρχείου. Το file descriptor του στέλνεται στην norm_exec. Η διεργασία γονέας αναλαμβάνει το κλείσιμο του αρχείου. Για εντολές χωρίς output redirection η norm_exec καλείται με όρισμα για file descriptor -1.

β) Σωλήνωση

Αρχικά είχε ειπωθεί ότι στη γραμμή εισόδου μπορεί να έχουμε περισσότερες εντολές, οι οποίες χωρίζονται με ‘;’. Η κάθε μία από τις εντολές αυτές όμως μπορεί να χρησιμοποιεί σωλήνωση, δηλαδή και αυτές μπορούν επίσης να χωριστούν σε περισσότερες. Η μόνη διαφορά εδώ είναι ότι η έξοδος μίας εντολής είναι η είσοδος της επόμενης. Για να επιτευχθεί αυτό, χρησιμοποιείται η συνάρτηση pipe(). Αφού κάνουμε ένα γραμμικό πέρασμα της εισόδου για την εύρεση του πλήθους των ‘|’, φτιάχνουμε ίσο αριθμό από pipes (δηλαδή για “cat in.txt | sort | head” φτιάχνονται 2 pipes). Τώρα η εντολή, με χρήση της strtok(), χωρίζεται με ‘|’ και παίρνουμε σειριακά κάθε εντολή (πάνω στις οποίες γίνεται ο έλεγχος για redirections που περιγράφτηκε προηγουμένως). Κάθε εντολή έχει έναν index, δηλαδή η “cat in.txt | sort | head”, αφού χωριστεί γίνεται:

0: cat in.txt

1: sort

2: head

Με αυτό γνωρίζουμε αν βρισκόμαστε στην πρώτη, στις μεσαίες ή στην τελευταία εντολή, πράγμα που είναι σημαντικό για τη σωστή διαχείρηση των pipes. Μετά ακολουθεί η norm_exec. Πρίν εξηγηθεί περαιτέρω, σημαντική είναι η κατανόηση της norm_exec:

Η norm_exec δέχεται εντολή, η οποία χωρίζεται με “ “ όπου κάθε κομμάτι εκχωρείται σε ένα vector, το οποίο χρησιμοποιείται ως όρισμα για την execvp. Αν το fd ειναι διάφορο του -1 σημαίνει ότι η εντολή έχει redirection και η έξοδος της εντολής πρέπει να εισαχθεί στο αρχείοκαι όχι στο stdoutput. Αυτό επιτυγχάνεται με την χρήση της dup2, μια πολύ χρήσιμη συνάρτηση, η οποία είναι ακραία για σωλήνωση.

Αφού κληθεί η norm_exec και γνωρίζοντας σε ποιά εντολή βρισκόμαστε, εφαρμόζουμε ανάλογα dup2:

Για compos = index εντολής

- Αν compos == 0, βρισκόμαστε στην πρώτη εντολή, που σημαίνει ότι θέλουμε μόνο να γράψουμε στο pipe και όχι να διαβάσουμε.

- Αν compos == αριθμό των ‘|’, βρισκόμαστε στην τελευταία εντολή, άρα μόνο διαβάζουμε είσοδο και έξοδο στο stdoutput.

- Αλλιώς, βρισκόμαστε στις ενδιάμεσες εντολές, δηλαδή διαβάζουμε από το pipe και γράφουμε στο pipe

Γενικά, το pipe από ποιό οποίο διαβάζουμε και στο οποίο γράφουμε διαλέγεται ως εξής:

Για την nth εντολή:

- διάβασμα από το pipe[read] n-1

- γράψιμο στο pipe[write] n

γ,ζ) Background & Signals

Τα signals ^C και ^Z αγνοούνται από το κέλυφος με την χρήση sigaction. Για την διεργασία παιδί ^C, ^Z έχουν τις Προκαθορισμένες τιμές και δεν αγνοούνται όσο τρέχει η εντολή. Αν όμως βρεθεί '&' στο τέλος της εντολής, τότε το παιδί θεωρείται πως είναι για background και επίσης αγνοεί τα σήματα αυτά, στην περίπτωση αυτή όμως ο γονέας δεν περιμένει να τερματήσει το παιδί πρωτού λάβει την επόμενη γραμμή εντολής.

δ) Wild Characters

Για τους wild characters καλείται η wildch_matc μέσα στην norm_exec, η οποία χρησιμοποιεί την glob για την εύκολη και αποτελεσματική εύρεση του αποτελέσματος.

η) myHistory

Για την myHistory αρχικά χρησιμοποιείται ο τύπος δεδομένων myHistory, ο οποίος περιέχει έναν πίνακα με τις 20 τελευταίες γραμμές εντολών, δείκτης στην πιο πρόσφατη εντολή και τρέχον μέγεθος.

Η αποθήκευση των εντολών στον πίνακα γίνεται κυκλικά, δηλαδή αρχικά γεμίζει σειριακά και όταν γεμίσει ο πίνακας, η εισαγωγή ξεκινάει πάλι από τον πρώτο κόμβο (τον πιο παλιό).

Κάθε εντολή ελέγχεται εαν είναι κλήση της myHistory, με τη χρήση της hist_call(). Εάν έχει κληθεί χωρίς δείκτη, εκτυπώνεται ο πίνακας, αλλιώς καλείται η hist_get() η οποία επιστρέφει κάποια εντολή, ανάλογα με τον δείκτη.

Αλλιώς η εντολή αποθηκεύεται στον πίνακα με την hist_update().

	γενικα σχόλια	ΝΑΙ	ΜΕΡΙΚΩΣ	ΟΧΙ
shell	Βασική λειτουργία κελύφους εισαγωγής και κλήσης απλών εντολών πχ ls		X	
	επιστροφή και εκτύπωση αποτελεσμάτων	X		
redirections	μονή ανακατευθυνση > ή < (5)	X		
	διπλή ανακατεύθυνση	X		
	ανακατεύθυνση προσθήκης >>	X		
pipes	απλό pipe cat file |grep "nikos" (5)	X		
	Συνδυασμός με ανακατεύθυνση cat file |grep "nikos" >file2	X		
background execution	εκτέλεση εντολών στο background	X		
	Εκτελεση πολλων εντολών σε μία γραμμή sort file1 &; ls &;	X		
	Επιστροφή ολοκλήρωσης πισω στο shell	X		
wild chars	υποστηριξη * (για τρέχοντα κατάλογο μονο)	X		
	υποστήριξη ? (για τρέχοντα κατάλογο μονο)	X		
aliases	create και χρήση alias θα πρέπει να το φτιαξουν εξ αρχής την δομή και πριν την εκτέλεση κάθε εντολής θα πρέπει να ελέγχουν τη δομή αυτή)			X
	destroy (διαγραφή)			X
history	myHistroy save and print	XX		
	κληση εντολής χωρίς επαναπληκτρολόγηση (myhistory 5)			
signals	control-C α) σκοτώνει την εσωτερική διεργασία	X		
	control-C β) δεν επηρεάζει το κέλυφος	X		
	control-Z α) σκοτώνει την εσωτερική διεργασία	X		
	control-Z β) δεν επηρεάζει το κέλυφος	X		