

ΟΝΟΜΑ: Πέτρος
ΕΠΩΝΥΜΟ: Βασιλόπουλος
Α.Μ.: 1115201000218

Η εργασία υλοποιεί το `jms` system χρησιμοποιώντας `named-pipes` για την σύνδεση του `jms_coord` με τον `jms_console` και επίσης `named-pipes` για την σύνδεση του `jms_coord` με τα `pools` τα οποία στην συνέχεια δημιουργούν παιδιά με την `fork` τα οποία αφού δημιουργήσουν φακέλους που θα περιέχουν τα αρχεία `stderr` και `stdout`, εκτελούν με `execvp` το `command` που τους δόθηκε.

Πιο συγκεκριμένα: Τρέχουμε το `jms_coord` το οποίο περνάει τα ορίσματα που πρέπει να δοθούν και τα `jms_in`, `jms_out` τα οποία αν στο `jms_console` έχουν δοθεί με `-w jms_in -r jms_out` στον `coord` είναι ανάποδα όπως ήθελε η άσκηση. Τα ορίσματα μπαίνουν μια οποιαδήποτε σειρά και αν δεν δοθούν όσα πρέπει τότε εμφανίζει μήνυμα λάθους. Στην συνέχεια, το `jms_coord` φτιάχνει το φάκελο με το `path` που του δόθηκε και έπειτα διαγράφει ότι `names-pipes` έχουν μείνει ανοιχτά από άλλες εκτελέσεις με την συνάρτηση `disconnect` η οποία κάνει `unlink` τα `fifo`. Μετά δημιουργεί τα `named-pipes` και τα `file descriptors` και περιμένει μέχρι ο `jms_console` να τα ανοίξει.

Ο `jms_console` αφού καταχωρεί τα ορίσματα με οποιαδήποτε σειρά ανοίγει τα `named-pipes` και μπαίνει σε ένα `loop` και διαβάζει από το αρχείο μία εντολή την φορά με την συνάρτηση `get_command` και στην συνέχεια με την `analyze` σπάει την εντολή σε `tokens` και συγκρίνει ποιά εντολή απο τις 10 έχει δοθεί, έπειτα γράφει την εντολή στο `named-pipe` για να την διαβάσει το `jms_coord` και μπαίνει σε μια `loop` που περιμένει να διαβάσει την απάντηση του για να προσωρήσει στην επόμενη εντολή.

Τα `named-pipes` ανοίγονται με `O_NONBLOCK` στην `open` για `read` απο τα δύο προγράμματα.

`Submit`: Το `jms_coord` παίρνει την εντολή απο το `named-pipe` το σπάει σε `tokens` και αν είναι `submit` κοιτάει αν είναι η πρώτη φορά που κάνει `submit`, αν ναι φτιάχνει ένα `pool` και στην συνέχεια για το παιδί `pool` καλεί την `pool_process` και περιμένει 1 sec μέχρι το `pool` να ανοίξει τα `fifo`, μετά κλείνει ότι `fd` έχει ανοίξει πιο πριν για να μην είναι πολλά ανοιχτά και ανοίγει αυτά που πρέπει. Ένας μετρητής `count` αυξάνεται για όταν φτάσει τα `max_jobs` να ανοίξει άλλο `pool` γράφει στο `pool` την εντολή και περιμένει να διαβάσει την απάντηση του και να την προωθήσει στον `jms_console`.

`status`: Βρίσκει σε ποιο `pool` ανοίκει το `jobid` που του δόθηκε καθώς κοιτάει σε κάθε επανάληψη αν το `jobid` είναι μεγαλύτερο από το `max_jobs` και αυξάνει ένα μετρητή `j` κάθε φορά, αν ναι τότε το `jobid` βρίσκεται το `pool[j]` αλλιώς κάνει `max+=max` ξανά ελέγχει. Στην συνέχεια κοιτάει αν υπήρξε πρόβλημα στο να ανοίξει τα `pipes` ή να γράψει σε αυτά καθώς αυτό σημαίνει ότι το `pool` έχει τερματίσει και τότε επιστρέφει σαν απάντηση στο `pool` ότι και το `job` έχει τερματίσει.

status-all: Κάνει ακριβώς ότι η status μόνο που στέλνει σε όλα τα pools το μήνυμα status-all και κάθε φορά που θα παραλάβει απαντήσεις ίσες με max_jobs αλλάζει pool για να πάρει και απο εκεί απαντήσεις. Άμα το pool έχει τερματίσει εμφανίζει για max_jobs επαναλήψεις ότι η κάθε job είναι finished.

show-active: Κάνει το ίδιο ακριβώς με status-all μόνο που στέλνει μόνο ότι jobs είναι active, σταματάει να διαβάζει μόνο όταν το pool του στείλει end.

show-finished: Κάνει το ίδιο ακριβώς με show-active μόνο που στέλνει μόνο ότι jobs είναι finished.

show-pools: Έχει μια επανάληψη που σπάει όταν έχουμε λάβει μηνύματα απο όλα τα pools και διαβάζει μια φορά απο κάθε pool. Αν το pool έχει τελειώσει στέλνει μήνυμα ότι το συγκεκριμένο pool δεν εξυπηρετεί κανένα job.

suspend: Βρίσκει σε ποιο pool υπάρχει το jobid και στέλνει το μήνυμα suspend jobid σε αυτό το pool, το pool ελέγχει αν το jobid είναι active και αν ναι στέλνει sigstop με το kill σε αυτό. Στην συνέχεια διαβάζει μέχρι να πάρει απάντηση ότι το pool έγινε suspended.

resume: Κάνει ακριβώς ότι η suspend μόνο που το pool στέλνει SIGCONT στο job.

shutdown: Στέλνει σε όλα τα pool SIGTERM τα οποία με κατάλληλο signal handler όταν πάρουν αυτό το σήμα κάνουν true μια flag στη συνέχεια το coord στέλνει το μήνυμα shutdown και το pool αν λάβει αυτό το μήνυμα κοιτάει αν το flag είναι true και τότε εκτελεί το shutdown που στέλνει σε κάθε job SIGTERM αν είναι active, ενώ αν είναι suspended SIGKILL γιατί μόνο έτσι μου τερμάτιζε.

Το script εκτελεί το list κάνοντας ls -l στον φάκελο που δόθηκε, το size κάνοντας cd στον φάκελο που δόθηκε και στην συνέχεια ls -slr και purge κάνοντας find για το όνομα jms στον φάκελο που βρίσκεται το script και στην συνέχεια rm. (δεν μπόρεσα να το κάνω για το φάκελο που θα δοθεί καθώς εκεί το cd δεν μου δούλεψε.