

Sistemas operativos

Práctica 3: MiniShell

Índice de Contenidos

Sistemas Operativos, Práctica 3.....	1
Descripción del código.....	2

Descripción del código

Para empezar tenemos una serie de funciones básicas, que comprueban si hay background o si se ha introducido el comando “cd”. También contamos con los métodos para redireccionar las salidas y las entradas, que básicamente son iguales salvo que en el de la entrada abrimos el fichero en modo lectura y en el de salida creamos un fichero nuevo con los permisos pertinentes. Contamos también con un método para comprobar si lo introducido es realmente un comando con la función “e_comando” donde usamos la llamada a sistema “fork”.

La función más importante de nuestro código es “e_comandos” en la cual creamos los pipes que necesitaremos. Para ello usamos una matriz dinámica con tantas filas como pipes necesite (que será el número de comandos -1). Recorremos cada fila de la matriz y en cada fila se reserva el espacio suficiente para guardar dos enteros, que se corresponderán con un 0 o con un 1 (lectura o escritura) y abrimos una “tubería” para comunicar los comandos que vayamos a utilizar.

También reservamos memoria para los procesos hijos, que será igual al número total de comandos que se pasen. Hacemos la comprobación de errores y probamos las redirecciones. Y para terminar contemplamos el caso de que haya más de un comando. En tal caso en el primer comando se duplica el valor de escritura de esa casilla del pipe y se pasa por salida estándar, si es el último comando se duplica el valor de lectura del pipe anterior y se pasa por entrada estándar y en caso contrario todo sigue con normalidad (salida del anterior por entrada del siguiente).