

# Algoritmo de la utilidad util4

1. Declarar variables necesarias, incluyendo un arreglo de estructuras ``users``, variables de control y manipulación de tuberías, y variables para el proceso hijo y el búfer de lectura.
2. Inicializar el arreglo ``users`` con valores predeterminados.
3. Crear una tubería utilizando la función ``pipe`` y verificar si se pudo crear correctamente.
4. Crear un nuevo proceso hijo utilizando la función ``fork`` y verificar si la creación fue exitosa.
5. En el proceso hijo:
  - a. Cerrar el extremo de lectura no utilizado de la tubería.
  - b. Redirigir la salida estándar (stdout) al extremo de escritura de la tubería utilizando ``dup2``.
  - c. Cerrar el extremo de escritura original de la tubería.
  - d. Ejecutar el comando "last" utilizando ``execlp``.
  - e. En caso de error en ``execlp``, imprimir un mensaje de error y salir con un código de error.
6. En el proceso padre:
  - a. Cerrar el extremo de escritura no utilizado de la tubería.
  - b. Abrir el extremo de lectura de la tubería como un flujo de archivo utilizando ``fdopen``.
  - c. Leer líneas del flujo de archivo y procesarlas.
  - d. Actualizar el arreglo ``users`` con la cuenta de usuarios.
  - e. Cerrar el flujo de archivo.
  - f. Esperar a que el proceso hijo termine utilizando ``wait``.
7. Iterar sobre los argumentos de línea de comandos y buscar los usuarios correspondientes en el arreglo ``users``.
8. Imprimir el nombre del usuario y su cuenta correspondiente.
9. Retornar un valor indicando una finalización exitosa del programa.