

SISTEMAS OPERATIVOS

Condiciones de competencia (carrera)

Elaborado por: Ukranio Coronilla

El objetivo de esta práctica es observar condiciones de competencia y las inconsistencias que se pueden producir, e implementar una solución sencilla que garantice la exclusión mutua.

Primero debe declarar una variable entera que comparten (en memoria compartida) un proceso padre y un proceso hijo y cuyo valor inicial es cero. El proceso padre será responsable de leer la variable y posteriormente incrementarla una unidad, mientras que el proceso hijo leerá la variable y posteriormente la decrementará en una unidad. Ambos procesos trabajarán de manera concurrente.

Para determinar si se ha presentado alguna inconsistencia se les debe proporcionar (tanto al proceso padre como al proceso hijo) en la línea de comandos el número N. Este valor indica el número de incrementos y decrementos que cada proceso debe realizar. Al finalizar ambos procesos deberá imprimirse el valor final de la variable compartida, la cual debería ser de cero en el caso de que todo haya salido bien.

Vaya probando con valores de N cada vez mayores hasta que se produzcan inconsistencias y determine a partir de qué valor de N suceden las inconsistencias.

Posteriormente y en otro programa, se debe resolver esta inconsistencia mediante la solución de Peterson(página 123) o la de alternancia estricta(página 121) la que se desee.

Importante:

Recuerde que al crear un proceso hijo no se duplican los recursos compartidos (archivos abiertos, memoria compartida...), por lo que solo es necesario reservar al inicio del código, la memoria compartida y después crear el proceso hijo. Así proceso hijo y proceso padre podrán acceder a los recursos compartidos.