

Tarea Programada 1: Comunicación entre varios procesos

Objetivo: poner en práctica la comunicación de varios procesos por medio de buzones.

Instrucciones: debe crear un programa en C o C++ que utilice llamados al sistema en Linux que reciba un documento a trabajar entre varios procesos.

El programa debe contar la cantidad de líneas que contiene el documento y la cantidad de caracteres sin contar los espacios en blanco. El documento debe ser procesado utilizando distintos procesos creados por medio del llamado al sistema fork.

Como parte de su entrega explique si es posible la comunicación de procesos utilizando solamente una región de memoria compartida en lugar de utilizar los buzones.

Notas:

- El programa DEBE correr en Linux, es decir, no se pueden utilizar llamados al sistema de Windows.
- El documento debe estar en una sección de memoria compartida, para ello utilice mmap. Tome este enlace como guía: <https://man7.org/linux/man-pages/man2/mmap.2.html>
- Se deben utilizar `msgrcv` y `msgsnd` de Linux, los puede revisar en este enlace: <https://linux.die.net/man/2/msgsnd>
- Si necesita realizar sincronización, cree su propia clase Semáforo. Utilice este enlace como referencia: <https://man7.org/linux/man-pages/man2/semget.2.html>
- Va a necesitar un "KEY" para ese semáforo, defínalo utilizando su carné estudiantil
- Los procesos solamente se pueden crear con fork, no se pueden utilizar otras estructuras.
- Debe trabajar en clases separadas. No se permite que todo el código esté en una sola clase o main.
- El trabajo es en parejas o individual.

Entregables:

- Diseño general, debe incluir posibles clases, métodos y un diagrama general e implementación de sus clases Semáforo y Buzón: 2 de octubre
- Entrega completa final: 16 de mayo

Valor: 10%

- Diseño y clases Buzón y Semáforo: 4%
- Entrega final: 6%