

<p>Tema 2</p> <p>CRUCE</p> <p>Vencimiento Martes 21 Julio 18:00 hs</p>	<p>75-08 Sistemas Operativos 2020 1er Cuatrimestre Trabajo Práctico 2</p>
<p>Nota: Toda consulta acerca del TP2 debe estar encabezada con las palabras TP2 en el subject y dirigida a so7508@gmail.com</p> <p>Pueden usarse para resolver el problema el constructor de un semáforo y los métodos wait() y post(). El acceso al valor interno del semáforo solo puede usarse para debug.</p>	

Grupo 2

90928 ALVAREZ, NATALIA NAYLA natalia.nayla.alvarez@gmail.com

68180 BONINO, ADRIAN GUSTAVO abonino@la.logicalis.com

96029 WALTER, FACUNDO IVAN facuwalter7@gmail.com

105463 Troucan-Jouve, Clement clem.tj@gmail.com

Programar en C++ los programas necesarios para poder correr el escenario descrito a continuación. **Cada programa deberá tener en su fuente e imprimir al lanzarse el nombre y padrón de los integrantes del grupo.** Durante la ejecución se preverán puntos de control para verificar el avance paso a paso de los mismos mediante lecturas desde la terminal. Deberán usarse los nombres y parámetros descritos en el enunciado (los nombres están en **negrita**). Deberán usarse las primitivas de mensajes System V que se encuentran en la página del curso. Los programas se compilarán con el comando *make all* y los ejecutables se borrarán con *make clean*. (o sea, debe programarse un Makefile adecuado)

Una soga por sobre un precipicio es usada por los monos para cruzar. La soga solo soporta el peso de *m* monos y solo puede ser usada en una dirección a la vez. Mientras que no haya monos en espera en sentido contrario y la capacidad de la soga lo permita, los monos pueden seguir cruzando. Un mono, si embargo, no debe ver impedido (pospuesto indefinidamente) su cruce por un arribo continuo de monos en sentido contrario

Deben programarse un **simio** genérico que recibirá su nombre y dirección (rumbo de cruce) por parámetro (se activará manualmente un número indeterminado de ellos). Los simios deben permitir la ejecución paso a paso. Un **inicializador** y un **finalizador** para comenzar y terminar la simulación y un **terminador** que se encarguen de las estructuras de IPC. Puede ser necesario un **coordinador**. Entregar todos los archivos en un subdirectorío con el nombre del tema (cruce) y comprimido (tgz o zip).