Pareja 3: Victoria Pelayo e Ignacio Rabuñal.

Explicación de nuestro programa:

1. Primero empezamos implementando la parte de la carrera de caballos.

Creamos carrera.c (donde está el main de todo el programa) y caballos.c y caballos.h para implementar ahí funciones que utilizaremos en el main.

En carrera.c primero creamos tantos procesos hijos como caballos debe haber.

Anteriormente hemos creado tantas tuberías como caballos para poder comunicarnos con ellos.

El padre se encargará de ir escribiendo en la tubería el tipo de dado que deben utilizar los caballos. Para ello utiliza la función “calcular\_tirada()” que devolverá el dado de siete caras si va el primero, el doble dado si va el último y si no el dado normal; si hay algún error devolverá -1.

Una vez que la carrera es comenzada tiene que ir leyendo de la cola de mensajes las tiradas de los caballos para ir actualizando su posición.

Cuando un caballo sobrepase la meta se enviará una señal a todos los caballos apra que acaben la carrera.

Los caballos esperarán a recibir una señal. Cuando la reciban leerán de la tubería el tipo de dado que tienen que usar. Con la función caballos se calcula cuál es la tirada del caballo y se escribe en “mensaje” la tirada y el id y se mete en la cola de mensajes, para que el padre lo lea y actualice posiciones y el estado de la carrera, si es necesario.

1. Después creamos el gestor de apuestas. El proceso principal ahora lo primero que hará será crear un proceso hijo, el gestor.

El gestor primero inicializará las apuestas, con la función “inicializar\_gestor()”.

Una vez todo inicializado, creará tantos hilos como ventanillas fueron indicadas, estos hilos harán la función ventanilla.

Después se quedará en estado de espera hasta recibir una señal. Cuando la reciba cambiará el estado de la carrera a EN\_PROCESO y esperará a que los hilos terminen antes de hacerlo él.

Función ventanilla: Primero establece el id correspondiente a la ventanilla, controlado mediante semáforos para que cada ventanilla tenga su id. Reservamos memoria para compartida.

Mientras la carrera no haya empezado, cada hilo va recibiendo mensajes y actualizando las apuestas. Todo esto lo van escribiendo en un fichero, esta escritura también esta controlada mediante semáforos, la actualización de “compartida” también lo está.

Cuando la carrera ha empezado terminan los hilos.

1. Después del gestor, el proceso principal creará otro hijo que será el apostador. El proceso apostador irá enviando apuestas, a la cola de mensajes, cada segundo durante la espera de que comience la carrera.

El padre primero dormirá el tiempo correspondiente, tiempo para que se realicen las apuestas, y finalmente enviará una señal al apostador para que se acaben las apuestas y otra al gestor para que cambie el estado de la carrera. El proceso principal esperará a que acaben estos dos procesos y finalmente dará comienzo a la carrera de caballos.