

**Lucía Peñaranda Pardo NIA 100068468**

**Rafael León Miranda NIA 100275593**

**Profesor: M. García Valls**

**8 de marzo de 2015**

## Laboratorio 3

### Hebras POSIX: gestión básica y sincronización

**Explique brevemente cómo se sincronizan `h_prod` y `h_rentab` entre ellas para acceder a los vectores `coefs` y `cuentas`.**

Ambas hebras han sido programadas para que nada más empezar a ejecutar su código se vayan a dormir, esperando a la misma variable de condición (`coef_update_cv`). La hebra que ejecuta la función `h_update`, cuando termina de actualizar el vector `coefs` despierta a `h_prod` y `h_rentab` mediante `broadcast`, de tal forma que, ambas competirán por el `mutex` (`coefs_m`), a partir de ese momento, acceden en exclusión mutua al vector `coefs` y al vector de `cuentas` para saber qué valores han de usar y después grabar los resultados de los cálculos en cada tipo de comisión dentro de la cuenta correspondiente.

Hacemos uso también de una variable global de tipo entero (`update_ok`) para que se cumpla la condición de salida de la espera y así despierten tanto `h_prod` como `h_rentab`.

**Explique cómo se sincronizan `h_prod` y `h_rentab` entre sí y con `h_total` y qué mecanismo ha utilizado para conseguir que lo hagan adecuadamente.**

Hemos añadido dos variables globales de tipo entero para usarlas como flags. Estas variables son: `rent_ok` y `prod_ok`. Como ambas hebras deben haber terminado su ejecución para despertar a `h_total` y la ejecución no lleva siempre el mismo orden sino que depende de cuál de las dos haya conseguido antes el `mutex` `coefs_m`, lo que queremos conseguir es que cada hebra sepa si la ejecución de la otra hebra ya ha tenido lugar para, de esta forma, saber si es ella la encargada de despertar a `h_total` o no.

Tenemos a parte de estos dos flags una variable de condición (`cuentas_cv`) con su correspondiente mutex (`cuentas_m`). Lo primero que hace `h_total` en su código es ir a dormir esperando que la última en ejecutar entre `h_prod` y `h_rentab` le despierte con un signal sobre la variable de condición (`cuentas_cv`).

**Explique brevemente cómo se sincroniza `h_total` con `h_update` y cómo lo consiguen, con qué mecanismo.**

Una vez que la hebra `h_update` haya acabado de actualizar los valores del vector de coeficientes entra a dormir esperando que le despierten con un signal sobre la variable de condición `fin_calculo_cv`. La señalización a esta variable sólo es realizada por la hebra `h_total` una vez concluido el cálculo de las comisiones totales que modifica el valor del campo correspondiente en el vector de cuentas corrientes.