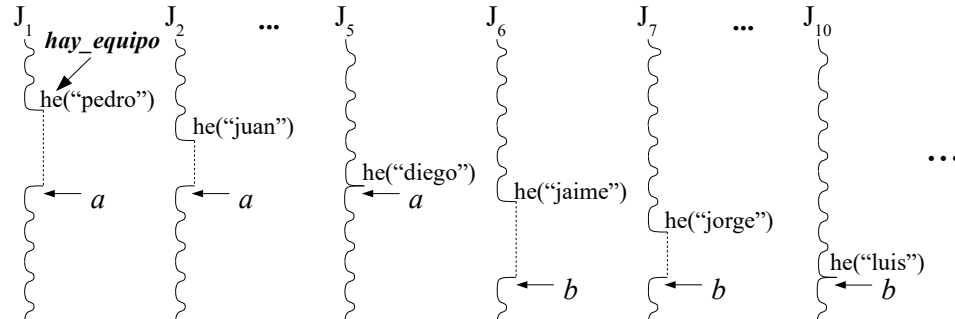


CC4302 Sistemas Operativos – Tarea 1

Semestre Primavera 2020 – Prof.: Luis Mateu

Parte a.- (2 puntos) Se necesita formar equipos de 5 jugadores de futbolito. Los jugadores son representados por threads que invocan la función *hay_equipo* indicando como argumento su nombre. Esta función espera hasta que 5 jugadores hayan invocado *hay_equipo* retornando un arreglo de 5 strings con los nombres del equipo completo. Ejemplo de uso:



En donde *a* es un arreglo con los nombres “pedro”, “juan”, ..., “diego”, y *b* es un arreglo con los nombres “jaime”, “jorge”, ..., “luis”.

La siguiente implementación de *hay_equipo* funciona casi siempre, pero es incorrecta:

<pre>char **equipo; // = nMalloc(5*sizeof(char*)); equipo en formación int k= 0; // cuantos jugadores han llegado nSem m; // = nMakeSem(1); para garantizar la exclusión mutua nSem tickets; // = nMakeSem(0); para hacer esperar hasta 4 jugadores</pre>	
<pre>char **hay_equipo(char *nom) { nWaitSem(m); // inicio secc. crítica equipo[k++]= nom; char **miequipo= equipo; if (k!=5) { // aún no hay equipo nSignalSem(m); // fin secc. crít. nWaitSem(tickets); // espera } else { // hay equipo: despertar a los 4 jugadores en espera</pre>	<pre> for (int i= 0; i<4; i++) nSignalSem(tickets); // preparar nuevo equipo equipo= nMalloc(5*sizeof(char*)); k= 0; nSignalSem(m); // fin secc. crít. } return miequipo; }</pre>

Haga un diagrama de threads que muestre una situación en donde un jugador recibe como resultado un equipo que todavía no se forma. Considere que en nSystem los tickets de un semáforo se otorgan por orden de llegada siempre.

Parte b.- (4 puntos) Programe una solución de *hay_equipo* que sea correcta y eficiente usando los semáforos de nSystem. *Hint*: un jugador debe crear su propio semáforo cuando necesita esperar, colocando su dirección en un arreglo de 4 semáforos. Al despertarse, el jugador destruye ese semáforo. Concretamente se pide programar las funciones *hay_equipo* e *init_equipo* en el archivo *equipo.c*. En la función *init_equipo* haga las inicializaciones de las variables globales. Su tarea debe pasar todos los tests de prueba incluidos en *test-equipo.c*.

Archivos

Descargue de U-cursos el archivo *t1.zip*. Contiene los archivos (i) *Makefile* que le servirá para compilar su tarea (Ud. debe definir la variable de ambiente NSYSTEM con la ubicación de nSystem en su computador), (ii) *equipo.h* con los encabezados de las funciones solicitadas, (iii) *test-equipo.c* con los tests de prueba y (iv) *equipo.c.plantilla* con una plantilla para programar *equipo.c*.

Entrega

Ud. debe entregar mediante U-cursos un *.zip* con el archivo *equipo.c*. y otro archivo en el formato de su elección con la solución de la parte a (puede ser una foto legible de su solución en papel). Se descontará medio punto por día hábil de atraso. Ud. puede recibir ayuda en la parte b para resolver problemas de depuración, pero no puede recibir ningún tipo de ayuda en la parte a.