

# CC3301 Programación de Software de Sistemas

## Tarea 4 – Semestre Otoño 2013

Prof.: Luis Mateu

En esta tarea Ud. debe programar un servidor y un cliente para un diccionario que acepta definiciones y consultas por Internet. El comando para lanzar el servidor se debe llamar `servidor_dict`, corre siempre en `localhost` y escucha a los clientes en el puerto 3000. El comando para lanzar un cliente se debe llamar `dict`. Para agregar una definición se usa el comando `dict def llave valor` y para consultar el comando `dict query llave`. Una definición se borra con el comando `dict del llave`. Su tarea debe funcionar conforme al siguiente ejemplo de uso:

<i>Shell 1</i>	<i>Shell 2</i>	<i>Shell 3</i>	<i>Shell 4</i>
% <b>./dict def A 1</b> ( <i>agrega la llave A</i> ) %			
	% <b>./dict query A</b> ( <i>consulta la llave A</i> ) 1 %		
	% <b>./dict query D</b> ( <i>espera</i> )		
		% <b>./dict query D</b> ( <i>espera</i> )	% <b>./dict query A</b> 1 %
% <b>./dict def A Uno</b> ( <i>redefine la llave A</i> ) %			
	( <i>se desbloquea</i> ) 4 %	( <i>se desbloquea</i> ) 4 %	% <b>./dict def D 4</b> %
% <b>./dict query AAA</b> ( <i>espera</i> )		% <b>./dict del D</b> ( <i>borra la llave D</i> ) %	% <b>./dict query A</b> Uno %
	% <b>./dict query D</b> ( <i>espera</i> )		
server has finished %	server has finished %		% <b>./dict shutdown</b>
% <b>./dict def BBB 3</b> no server %	% <b>./dict query D</b> no server		

Considere que el servidor fue lanzado previamente en un quinto shell. El tiempo avanza hacia abajo. Los comandos ingresados por el usuario se indican en **negritas**. Además se incluyen breves frases explicativas en letra *cursiva*, que no aparecen realmente en la pantalla.

Observe que cuando una llave no está definida, el servidor no entrega NULL, si no que se espera hasta que algún cliente defina un valor para esa palabra. Un cliente puede terminar el servidor con el comando `dict shutdown`, lo que hace además que terminen todos los clientes pendientes con el mensaje del ejemplo.

*Hint:* use en el servidor la implementación de los diccionarios que Ud. programó en la tarea 2 u opcionalmente la implementación vista en la clase auxiliar.

### Requerimientos

- Programe en C el servidor y el cliente, y además confeccione el Makefile que realiza la compilación.
- Debe usar sockets y threads para resolver el problema.
- Está prohibido usar cualquier forma de *busy-waiting* y además se deben evitar los *dataraces*.
- Debe enterrar legalmente cualquier thread lanzado.

### Entrega

Entregue un archivo .zip con los fuentes de su solución mediante U-cursos. Debe incluir el archivo Makefile. El plazo de entrega vence el Martes 2 de Julio a las 23:59. No se aceptarán tareas que no funcionen conforme al ejemplo de arriba. Se descontará medio punto por día de atraso.