

CS2S01 SISTEMAS OPERATIVOS

Semestre 2019-I

Laboratorio 3 (Parte II) 12 pts.

El objetivo de este laboratorio es familiarizarnos con los primitivos que existen a nivel de hardware para sincronismos dentro del OS.

Parte 1. 8pts

Genera un programa que genere 2, 4 y 8 hilos y que cada hilo este actualizando el valor de una variable global llamada “cuenta” en 1. Cada hilo debe aumentar esa variable global 50000 veces. Debe medirse el tiempo que demora la ejecución del programa para 2, 4 y 8 hilos para cada uno de los siguientes casos:

- a. Sin ningún tipo de sincronismo
- b. Utilizando pthreads_locks
- c. Utilizando el primitivo TAS
- d. Utilizando el primitivo TASTAS
- e. Utilizando el primitivo TASTAS con Backoff
- f. Utilizando el primitivo Ticket

Los primitivos se encuentran en el archivo atomic_ops.h que es una adaptación de los primitivos desarrollados en la librería Rochester Software Transactional Memory Systems (RSTM). Mayor información sobre los primitivos la pueden encontrar en:

<http://www.cs.rochester.edu/research/synchronization/pseudocode/ss.html> , para mas información revisar el paper de Michael Scott colgado en Canvas en la sección de papers.

Debe presentarse un gráfico del tiempo vs el numero de hilo y dentro del gráfico el tiempo que tomó cada primitivo.

Parte 2. 4pts

Para probar la capacidad de escalar de una barrera generar un programa que haga que 2, 4 y 8 hilos lleguen 10, 100, 1000, 10000 veces a una barrera. Hacer esto para

- a. Pthread_barrier
- b. Reverse sense barrier .

Presentar un gráfico del tiempo vs el numero de hilos para cada i , es decir 4 gráficos, comparando el tiempo de ejecución de ambas barreras.

Fecha de entrega: **Lunes 6 de mayo 2019. 11:59pm**

Enviar su archivo a rbustamante@utec.edu.pe

Desarrollar el código en C.