

UNIVERSIDAD
PANAMERICANA

Materia: Fundamentos de Programación en Paralelo

Profesor: Dr. Juan Carlos López Pimentel

Evidencia:
Thread States

Modo de trabajo: Equipo

Objetivo:

Consolidar el conocimiento sobre los diferentes estados de un *Thread* y su planificación mediante el desarrollo de un programa computacional gráfico que involucre el uso de varias ejecuciones concurrentes.

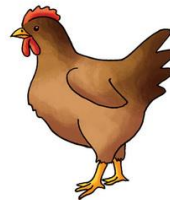
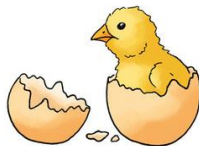
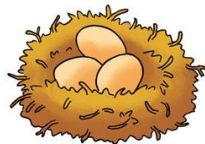
Instrucciones:

- Desarrollar un programa de computadora, que ilustre los diferentes estados en los que un *Thread* puede estar, desde su creación hasta su terminación, así como sus diferentes transiciones.
 - Los estados a ilustrar son los vistos hasta el momento NEW, RUNNABLE, TIMED WAITING and TERMINATED.
 - Transiciones para pasar a los diferentes estados como los mostrados previamente.
 - Transiciones intencionadas (*interruption*) para pasar de TIMED WAITING a RUNNABLE y de TIMED WAITING a TERMINATED.
- La ilustración puede ser mediante explicaciones en audio y/o imágenes.
- Las transiciones en los que cada Thread puede pasar deberán ser controlados por eventos de hardware (interrupciones) por ejemplo: mouse y/o teclado.

- El programa debe implementar en el código las dos formas de crear un Thread (por herencia y por interface con el lenguaje de programación Java)
- Tu programa deberás presentarlo en plenaria y tendrás 5 minutos para exponerlo y 3 minutos de preguntas y respuestas.
- Subir un documento PDF reportando con capturas de pantalla la parte visual, y un link donde se pueda visualizar todos los fuentes de tu programa. Anexar los integrantes de equipo que trabajaron.
- Recuerda que tu creatividad es muy importante.
 - Un ejemplo guía podría ser la simulación de movimiento aleatorio de dos diferentes vehículos; tu podrías agregarle aspectos audio visuales para ilustrar los diferentes estados



- Otro ejemplo guía podría ser el nacimiento alternativo de un pollito hasta ser gallina; similar al ejemplo previo, podrías agregarle aspectos audio visuales para ilustrar los diferentes estados



Nota: para completar la actividad anterior exitosamente tendrás que investigar el funcionamiento de tus propios componentes audio-visuales.

Evaluación:

Nota: Tiempo y forma es obligatorio.

	Créditos: Nombre y Logo de la escuela, Nombre de la Ingeniería, Nombre del Profesor, Nombre del Estudiante (s), Id, y Fecha. En alguna parte del programa ejecutado. Este aspecto restará puntos si no lo incluyes (10%)
20%	El programa presentado ilustra (mediante audios y/o imágenes) los diferentes estados en los que un <i>Thread</i> puede estar, desde su

	<p>creación hasta su terminación (al menos los vistos en clase).</p> <p>5 % NEW</p> <p>5 % RUNNABLE</p> <p>5 % TIMED WAITING</p> <p>5 % TERMINATED</p>
20%	Se implementa el uso de eventos (ejemplo: mouse, teclado, entre otros) para controlar las transiciones de los estados de los <i>Threads</i> .
25%	<p>El programa presenta las transiciones:</p> <p>5% New – Runnable</p> <p>5% Runnable – Timed Waiting</p> <p>5% Runnable – Terminated</p> <p>5% Timed Waiting – Terminated</p> <p>5% Timed Waiting – Runnable</p>
20%	El programa implementa en código las dos formas de crear un <i>Thread</i> (visto en clase)
10%	Presentación: Se ajusta a los tiempos especificados en su presentación y durante ese tiempo explica todos los aspectos realizados. Se le pregunta sobre lo que hizo y responde correctamente
5%	Reporte PDF
	Realiza auto y coevaluación. Dependiendo de esta actividad será el % que el alumno tendrá de participación e incidencia en la calificación.