

Scheduling en Tiempo Real

Sistemas Operativos Avanzados

Nicole Carvajal

Rubén González

Edisson López

Otto Mena

Cristina Soto

Tecnológico de Costa Rica
Maestría de Ciencias de la Computación
Semestre 1, 2021

Simulación del comportamiento de varios algoritmos de scheduling clásicos para Sistemas Operativos de Tiempo Real (RTOS). Con una interfaz gráfica hecha con GTK y generación de una presentación Beamer como salida.

Algoritmo: Rate Monotonic

- Propuesto por Liu y Layland (1973)
- Scheduling de tiempo dinámico
- Es óptimo.
- Expropiativo, de mayor prioridad primero
- La prioridad de una tarea es inversamente proporcional a su periodo

Algoritmo: Earliest Deadline First

- Propuesto por Liu y Layland (1973)
- Scheduling de tiempo dinámico real
- Es óptimo.
- Expropiativo, de mayor prioridad primero
- La prioridad de una tarea es inversamente proporcional al tiempo pendiente para que se dé su deadline

Algoritmo: Least Laxity First

- Propuesto por Leung (1989)
- Scheduling de tiempo dinámico real
- Es óptimo.
- Expropiativo, de mayor prioridad primero
- La prioridad de una tarea es inversamente proporcional a su laxity. El laxity de la tarea i , d es el deadline, c es el tiempo de computación y t es el momento en el tiempo, se calcula: $(L_i = d_i - t_i - c_i)$

Tests de Schedulability

Test de Liu y Layland

- Fórmula: $(\mu = \sum c_i/p_i \leq U(n) = n(2^{1/n} - 1))$
- Resultado del test RM: $(\mu = 1,25 > U(n) = 0,83)$, **rechazada**.
- Usando RM, puede que para este conjunto de tareas ocurra un incumplimiento del deadline o puede que no.
- Resultado del test EDF: $(\mu = 1,25 > 1)$, **rechazada**.
- Usando EDF, puede que para este conjunto de tareas ocurra un incumplimiento del deadline o puede que no.

Test de Bini

- Fórmula: $(\mu = \prod (c_i/p_i + 1) \leq 2)$
- Resultado del test: $(\mu = 2,62 > 2)$, **rechazada**.
- Puede que para este conjunto de tareas ocurra un incumplimiento del deadline o puede que no.





