

# Sistemas de Tempo-Real / Programação em Tempo-Real

## Segundo Trabalho Prático

Considere o seguinte conjunto de tarefas:

Tarefa	Computação	Deadline	Período
T1	3	7	8
T2	2	6	9
T3	3	15	15

Analisar a **escalonabilidade** para as políticas: Rate-Monotonic (RM), Deadline-Monotonic (DM) e Earliest Deadline First (EDF). O **deadline** só deve ser considerado na política Deadline Monotonic. Para as outras políticas, favor considerar que o deadline = período. Implementar as três políticas, no conjunto de tarefas acima, usando o RTAI (*Real-Time Application Interface*), tanto no nível de kernel quanto no nível de usuário.

Para esse trabalho o ideal é que, **mas não obrigatório**, cada tarefa seja implementada como uma **thread** (usando a função `pthread_create`). Mas pode ser implementada usando `rt_task_init` sem problemas. Quem implementar usando thread poderá ter, além dos pontos nesse trabalho, mais **dois pontos de bonus** em qualquer outro trabalho. Você terá que indicar qual a **unidade de tempo** que cada valor acima representa (milissegundo, microsegundo, segundo, etc). A passagem do tempo de execução pode ser implementada usando, por exemplo, a função `rt_sleep`. Adicionalmente, imprimir informações de **log**, por exemplo, para cada instância da tarefa, o instante de tempo que a tarefa começou a execução e quando a tarefa concluiu a execução. Para cada uma das três políticas e para cada forma de implementação (kernel e usuário) comentar sobre a **precisão** dos resultados a partir da avaliação dos logs (comentar sobre *jitters*, etc). **Sugestão:** execute diversas instâncias das tarefas até o MMC de todos os períodos.

O que deve ser entregue:

- 1) A análise de escalonabilidade para as três políticas: RM, DM e EDF;
- 2) Código das tarefas acima escalonadas pelas políticas RM, DM e EDF, implementadas no nível de kernel;
- 3) Código das tarefas acima escalonadas pelas políticas RM, DM e EDF, implementadas no nível de usuário;
- 4) Para cada uma das 3 políticas, comentar sobre a precisão dos resultados, a partir do log, considerando as duas formas de implementação (kernel e usuário);
- 5) Todos os logs das (seis) execuções.

Data de Entrega: **17 de novembro de 2008** (segunda).

Após esta data será descontado **1.0 ponto** por dia de atraso.

Enviar para os emails do professor e do monitor.

Alunos de graduação poderão fazê-la em **triplas**.

Alunos de mestrado deverão fazê-la **individualmente**.

Favor começar a fazer **imediatamente** para não correr risco de entregar atrasado.

Obviamente que o trabalho pode ser entregue antes.

**Não será dado nenhum prazo adicional.**