



# Utilização do *kernel* tempo-real FreeRTOS

## Objetivos:

Tomar contacto com o kernel tempo-real FreeRTOS.

1. Conhecer a API deste kernel, a estrutura das tarefas, as funções de inicialização e terminação das aplicações e os mecanismos de IPC.
2. Efetuar a programação e depuração de aplicações tempo-real usando o kernel FreeRTOS.

## Procedimentos:

### 1. *Análise do código fornecido*

Descarregar o software de apoio à sessão “freertos-class-sources.zip”, disponível na página da disciplina.

1. Observar o código fonte do módulo (“main.c”). Analisar as chamadas ao sistema para definição, criação e ativação de tarefas. Analisar a estrutura de uma tarefa.
2. Consulte o ficheiro README. Compile e execute o exemplo fornecido. Observe a execução concorrente das duas. Observe e explique os efeitos observados.

Nota: A execução do FreeRTOS em ambiente MSWindows é bastante útil do ponto de vista didático-pedagógico, pois simplifica consideravelmente o processo de desenvolvimento. Todavia, este ambiente degrada consideravelmente o comportamento temporal. Assim, deve ter em atenção que poderão ocorrer atrasos significativos nos instantes de ativação de alguns eventos. Em todo o caso o escalonamento das tarefas e o processamento dos eventos realiza-se na sequência correta.

### 2. *Tarefas a realizar:*

Realize as tarefas abaixo indicadas. Documente todas as alterações ao código que efetuar. Deverá entregar um relatório (2 páginas, máximo) com a análise dos resultados observados e indicação das alterações ao software introduzidas.

1. Inverta a prioridade das tarefas. Observe o padrão de ativações resultantes e anote eventuais diferenças relevantes, justificando-as.
2. O acesso ao terminal (escrita de strings) não é reentrante, pelo que numa situação genérica poderia ocorrer corrupção de informação. Modifique o programa fornecido



por forma a implementar um mecanismo do tipo “Gatekeeper”, i.e., para que a escrita de strings seja efetuado por uma única tarefa. O mecanismo de IPC deve ser uma fila.

3. **(Opcional)** Implemente um mecanismo de controlo de admissão. Este poderá ser suportado por uma função, a qual deverá receber todos os argumentos necessários para a criação da tarefa (i.e., os mesmos argumentos que a função `xTaskCreate()`, bem como eventuais parâmetros adicionais que julgue necessários).
  - Em função do resultado do teste de escalonabilidade, esta função deverá realizar a criação da tarefa invocando a função `xTaskCreate()`.
  - Assuma que a atribuição de prioridades à tarefas respeita o critério “Rate-Monotonic”

A API do FreeRTOS encontra-se documentada em <http://www.freertos.org/a00106.html>