Curitiba, 22, março de 2022.

**Disciplina**: Sistemas Computacionais

**Professor:** Jhonatan Geremias

**Curso: Engenharia de Computação**

**Nome Estudante: Tasi Guilhen Pasin**

**Atividade Prática / Relatório**

**Tarefas FreeRTOS**

# **Descrição da Atividade:**

Esta atividade é composta por duas etapas, primeiramente responder as questões do item 1, na sequência implementar o código especificado no item. A atividade compreende o conceito de tarefas e o seu uso no FreeRTOS.

**Entrega:**

Esta atividade deverá ser entregue até o dia **05/04/2022** no Canvas.

O estudante deverá entregar um arquivo “.pdf” contendo as respostas do roteiro de atividade item 1.

O item 2 é a implementação do código no FreeRTOS, seguindo a especificação do roteiro. O estudante deverá implementar o seu código apenas dentro do arquivo “example.c”. Entregar apenas o arquivo “example.c” onde foi codificado, deve conter o nome do estudante e curso adicionados no cabeçalho do arquivo como comentário.

Ambos os arquivos devem estar compactados no formato zip e postado no Canvas até a data limite.

**Roteiro da Atividade:**

## **Com apoio ao material fornecido responda:**

* 1. Descreva o uso da função xTaskCreate(), para que é utilizada essa função? Quais os parâmetros da função e para que são utilizados?

**R.:** Cria nova instância de uma tarefa. Os parâmetros são:

1. **pvTaskCode** -> A função que contém o conjunto de instruções da task;
2. **pcName** -> Nome para a task para facilitar o debug;
3. **usStackDepth** -> É a quantidade de memória alocada para a task;
4. **pvParameters** -> Parâmetro que pode ser de qualquer tipo. Vai da implementação da task saber o tipo correto que está sendo esperado para fazer o cast;
5. **uxPriority** -> Define a prioridade da task. Quanto maior o valor, maior a prioridade.
   1. Descreva para que são utilizadas as funções xTaskDelay() e vTaskDelete()?

**R.:** xTaskDelay() -> Passa a tarefa informada para o estado bloqueado por um período de tempo, que é informado na chamada da função;

vTaskDelete() -> Deleta uma instância de uma tarefa existente.

* 1. O que faz a função xTaskStartScheduler()?

**R.:** Inicia o agendador do FreeRTOS. Iniciar o agendador FreeRTOS joga a tarefa com maior prioridade criada para o estado "Executando"

1. **Implemente o programa no FreeRTOS conforme a especificação:**

Contexto: Para auxiliar no combate da Pandemia do Covid-19, o centro pesquisa de equipamentos médicos está convidando você a auxiliar no desenvolvimento de um monitor dos dados vitais dos pacientes. O monitor deve registrar os batimentos cardíacos do paciente (considerar entre 20 e 140 batimentos cardíacos - abaixo de 50 mensagem de batimento cardíaco alto, acima de 90 apresentar mensagem batimento cardíaco alto),

nível saturação do oxigênio (considerar oxigenação entre 80% e 100% - abaixo de 90% mensagem de saturação baixa) no sangue e

temperatura (considerar temperatura de 34° a 41° - abaixo de 35° apresentar mensagem de hipotermia, acima de 37,5° mensagem de febre).

* 1. Implementar um programa no FreeRTOS destinado a equipamentos médicos que deve medir os batimentos cardíacos, saturação de oxigênio e febre;
  2. Deverá ser criado três tarefas um para monitorar cada um dos dados vitais;
  3. A criação das três tarefas deve ser realizada no main\_;
  4. Para simular os dados vitais do paciente deverá ser utilizado funções randômicas para gerar cada um dos dados vitais;
  5. Os dados vitais (aleatórios) devem ser gerados dentro de cada tarefa;
  6. Todas as tarefas criadas devem ter a mesma prioridade;
  7. Efetuar a passagem de parâmetros para cada tarefa (sendo “Batimentos:”,” Saturação:” e “Temperatura:”);
  8. Fornecer os dados vitais na saída do console (batimento cardíaco, temperatura, saturação) – utilizar a função vPrintStringAndNuber();
  9. As tarefas devem gerar alertas quando os dados vitais dos pacientes estiverem alterados (mensagem no console).
  10. Utilizar a função vTaskDelay() configurando um tempo de um segundo para cada tarefa.
  11. Todas tarefas devem definir sua exclusão explicita utilizando a tarefa vTaskDelete();
  12. O código deve ser documentado, utilizar os comentários em toda a extensão do programa.