

# INF0992 - Programação Avançada em C

## Enunciado do Trabalho 2

Esse trabalho é composto de 5 exercícios sobre programação paralela.

Complete os arquivos `Ex[1-5].cs`, de forma que executem corretamente e gerem o resultado esperado.

O código deve estar legível e corretamente indentado, adicionando comentários onde necessário para a compreensão. É recomendado o uso do formatador automático de C# do VSCode. Note que a qualidade do código e dos comentários será levada em conta na avaliação.

Antes de começar o trabalho, execute `python3 gen_ints.py`. Isso gerará os arquivos auxiliares dos exercícios. Pode levar alguns segundos. Caso não tenha o Python instalado, os arquivos estarão disponíveis em um link do Google Drive.

Os 5 arquivos `.cs` devem ser submetidos no Moodle. Alternativamente, um arquivo `.zip` com os 5 arquivos pode ser submetido. O trabalho pode ser feito individualmente ou em duplas.

### Exercício 1 - 1/10 pontos

Imprima os números 1 - 1000 em uma thread, e os números 1001-2000 em outra thread. Use `System.Threading.Thread`. Quando ambas as threads terminarem, imprima “Pronto.”.

### Exercício 2 - 1/10 pontos

Imprima os números 1 - 1000 em uma thread, e os números 1001-2000 em outra thread. Use `System.Threading.Task`. Quando ambas as threads terminarem, imprima “Pronto.”.

### Exercício 3 - 2/10 pontos

Calcule e imprima o produto interno dos vetores A e B. A versão sequencial já está disponível; crie uma versão paralela. Para esse exercício, o resultado não precisa ser mais rápido que a versão sequencial.

### Exercício 4 - 3/10 pontos

Implemente a versão paralela da função `sequential_mergesort`. Para este exercício, pontos serão deduzidos se o resultado levar mais tempo para computar do que a versão sequencial.

Dicas: Não criar uma quantidade excessiva de tasks. Tentar criar tasks apenas quando o tamanho do problema for grande. Rodar com `dotnet run --configuration Release` para melhor desempenho.

### **Exercício 5 - 3/10 pontos**

Crie um programa paralelo que lê o arquivo `transactions.txt`, executa as transações, e imprime os valores finais corretamente, como em `expected_transaction_results.txt`. Para garantir o funcionamento correto, é preciso usar um lock para cada conta.