

## Mini EP 5

Este MiniEP tem como objetivo interpretar o tempo de execução de funções. Para isso, foram disponibilizadas duas funções: `eh_primo(n)`, que verifica se um número `n` é primo, e `soma_primos(n)`, que realiza a soma dos `n` primeiros números primos.

### Testes de tempo

Para testar o tempo de ambas as funções você deve utilizar o macro `@time` da linguagem Julia, que retorna o tempo em segundos que um código levou para ser executado.

Essa macro deve ser indicada antes da chamada da função, como feito no código-fonte fornecido.

Na primeira execução do `@time soma_primos()` a função `soma_primos()` é compilada, e isso faz com que o tempo de execução do primeiro teste seja maior. Por isso, quando você for testar, deve sempre deixar um primeiro caso teste de descarte, e esse resultado deverá ser desconsiderado.

Alguns casos de testes já estão disponíveis no arquivo, mas você pode modifica-los para entender melhor a relação entre o valor do parâmetros das funções e o tempo de execução das mesmas.

Caso você realize os testes direto no terminal de Julia, não é necessário realizar o descarte do primeiro teste e você pode retirar os casos teste indicados no arquivo. Você pode incluir as funções do arquivo no terminal de Julia com o comando:

```
julia> include("cf_miniep5.jl")
```

Além disso, se desejarem, podem também realizar otimizações no código de ambas as funções, como visto em aula, e verificar se há mudanças nos tempos de execução.

### Exemplo de uso

Ao executar o código-fonte fornecido, as primeiras linhas de resultado são:

```
> julia cf_miniep5.jl
```

```
Início dos testes da função eh_primo()
```

- Teste de descarte para função `eh_primo()`, `n=1`, resultado: `false`  
0.016618 seconds (12.71 k allocations: 949.703 KiB, 99.11% compilation time)
- Teste para `n=1`, resultado: `false`

```
0.000034 seconds (7 allocations: 416 bytes)
- Teste para n=11, resultado: true
```

O valor indicado mais a esquerda é o tempo em segundos que a função levou para ser executada para os parâmetros fornecidos, note que na primeira execução, o macro `@time` indica entre os parênteses que 99.11% daquele tempo foi utilizado para a compilação do programa. As demais informações dentro dos parênteses dizem respeito a alocação de memória, e não precisam ser consideradas para sua análise.

## Entrega

Você deve analisar o tempo de execução de ambas as funções, descrevendo a ligação desses tempos com o valor de `n` e com o algoritmo que está sendo utilizado.