

# SuperComputação

Aula 2 – Detalhes de implementação e desempenho

2020 – Engenharia

Luciano Soares <[lpsoares@insper.edu.br](mailto:lpsoares@insper.edu.br)>

Igor Montagner <[igorsm1@insper.edu.br](mailto:igorsm1@insper.edu.br)>



# Hoje

- Implementação vs algoritmo

# Implementação e Algoritmo

# Algoritmo

# Algoritmo

"Sequência finita de passos executáveis  
que resolve um problema"

# Implementação

# Implementação

"Transformação de um algoritmo em um programa executável"

# Quanto tempo um programa demora?

## Algoritmo:

- complexidade computacional (classes de algoritmos)
- estruturas de dados (abstratas)
- **provado matematicamente, não muda**

## Implementação:

- medido em segundos, para uma certa entrada
- tecnologia usadas (linguagens de programação, bibliotecas)
- hardware usado
  - Clock de CPU e RAM, tamanho do Cache
  - # de núcleos
- **imprecisão**





# Quanto tempo um programa demora?

## **SuperComputação começa quando Desafios de Programação acaba**

Já temos

- um bom algoritmo (complexidade computacional)
- estruturas de dados eficientes
- uma implementação inocente desse algoritmo



# Atividade prática

## Começando SuperComputação

1. Implementar algoritmos simples usando recursos de C++
2. Deixar nosso código minimamente organizado usando referências



# **Atividade prática**

**20 minutos para Parte 1 - Alocação de memória ....**



# **Atividade prática**

**30 minutos para Parte 2 e 3 ....**



# Atividade prática

**10 minutos para Parte 4 ....**

# Fechamento

Nossa otimização não funcionou, por que?

- Medir quanto tempo cada função demora?
- Nossa função ficou mais rápida? Se sim, quanto? Se não, por que?

# Insper

[www.insper.edu.br](http://www.insper.edu.br)