

# MAC0102 Caminhos no Bacharelado em Ciência da Computação

## Trilha de Teoria da Computação

Cristina G. Fernandes

`cris@ime.usp.br`

13 de abril de 2023

# Trilha de Teoria da Computação

Para receber o certificado dessa **trilha**, o aluno deve cursar

- ▶ as **obrigatórias** de pelo menos **dois módulos** e
- ▶ pelo menos 7 disciplinas da **trilha**.

# Trilha de Teoria da Computação

Para receber o certificado dessa **trilha**, o aluno deve cursar

- ▶ as **obrigatórias** de pelo menos **dois módulos** e
- ▶ pelo menos 7 disciplinas da **trilha**.

**Três módulos:**

- ▶ Algoritmos
- ▶ Matemática discreta
- ▶ Otimização

Quais são as obrigatórias dos módulos?

# Módulo Algoritmos

- ▶ MAC0328 Algoritmos em Grafos
- ▶ MAC0414 Autômatos, Computabilidade e Complexidade

# Módulo Algoritmos

- ▶ **MAC0328 Algoritmos em Grafos**
- ▶ **MAC0414 Autômatos, Computabilidade e Complexidade**
- ▶ MAC0325 Otimização Combinatória
- ▶ MAC0327 Desafios de Programação I
- ▶ MAC0345 Desafios de Programação II
- ▶ MAC0331 Geometria Computacional
- ▶ MAC0336 Criptografia para Segurança de Dados
- ▶ MAC0385 Estruturas de Dados Avançadas
- ▶ MAC0450 Algoritmos de Aproximação
- ▶ MAC0333 Armazenamento e Recuperação de Informação
- ▶ MAC0465 Biologia Computacional
- ▶ MAC0466 Teoria dos Jogos Algorítmica

# Módulo Matemática Discreta

- ▶ **MAT0206 Análise Real**
- ▶ **MAT0264 Anéis e corpos**
- ▶ **MAC0320 Introdução à Teoria dos Grafos**
  
- ▶ MAC0414 Autômatos, Computabilidade e Complexidade
- ▶ MAC0694/695 Combinatória I e II
- ▶ MAC0436/556 Tópicos de Matemática Discreta I e II
- ▶ MAC0691 Tópicos na Teoria Algébrica dos Grafos.
- ▶ MAE0221/224 Probabilidade I e II
- ▶ MAE0228 Noções de Probabilidade e Processos Estocásticos
- ▶ MAE0326 Aplicações de Processos Estocásticos
- ▶ MAT0311 Cálculo Diferencial e Integral V
- ▶ MAT0225 Funções Analíticas
- ▶ MAT0265 Grupos
- ▶ MAT0234 Medida e Integração

# Módulo Otimização

- ▶ MAC0315 Otimização Linear
- ▶ MAC0325 Otimização Combinatória
  
- ▶ MAC0300 Métodos Numéricos da Álgebra Linear
- ▶ MAC0343 Otimização Semidefinida e Aplicações
- ▶ MAC0427 Otimização Não Linear
- ▶ MAC0473 Otimização Inteira
- ▶ MAC0450 Algoritmos de Aproximação
- ▶ MAC0419 Métodos de Otimização em Finanças
- ▶ MAC0461 Introdução ao Escalonamento e Aplicações
- ▶ MAC0691 Tópicos na Teoria Algébrica dos Grafos
- ▶ MAC0452/552 Tópicos de Otimização Combinatória I e II
- ▶ MAC0418 Tópicos Especiais de Programação Matemática

# Trilha de teoria

## Esse semestre:

- ▶ **MAC0320** Introdução à Teoria dos Grafos
- ▶ **MAT0311** Cálculo Diferencial e Integral V
- ▶ MAC0327 Desafios de Programação
- ▶ MAC0336 Criptografia para Segurança de Dados
- ▶ MAC0345 Desafios de Programação Avançados
- ▶ MAC0427 Programação Não Linear
- ▶ MAC0436 Tópicos de Matemática Discreta I
- ▶ MAC0473 Otimização Inteira
- ▶ MAC0691 Tópicos na Teoria Algébrica dos Grafos

## Próximo semestre:

- ▶ **MAC0328** Algoritmos em Grafos
- ▶ **MAC0414** Autômatos, Computabilidade e Complexidade
- ▶ **MAC0315** Otimização Linear
- ▶ MAC0385 Estruturas de Dados Avançadas
- ▶ MAC0450 Algoritmos de Aproximação
- ▶ MAC0333 Armazenamento e Recuperação de Informação
- ▶ MAC0300 Métodos Numéricos de Álgebra Linear



# Mais sobre os módulos

## Três módulos:

- ▶ Algoritmos (hoje)
- ▶ Matemática discreta
- ▶ Otimização

## Semana que vem (20 de abril):

Guilherme e Marcel vão falar de matemática discreta e otimização.

# Escalonamento em máquinas idênticas

Dados:  $m$  máquinas

$t$  tarefas

duração  $d[i]$  da tarefa  $i$  ( $i = 1, \dots, t$ )

# Escalonamento em máquinas idênticas

Dados:  $m$  máquinas

$t$  tarefas

duração  $d[i]$  da tarefa  $i$  ( $i = 1, \dots, t$ )

um escalonamento é

uma partição  $\{M[1], \dots, M[m]\}$  de  $\{1, \dots, t\}$

# Escalonamento em máquinas idênticas

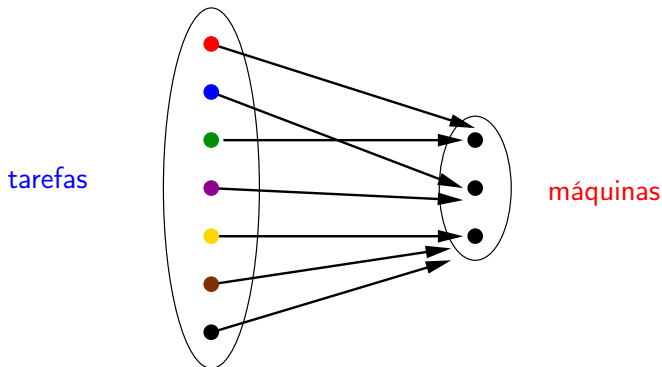
Dados:  $m$  máquinas

$t$  tarefas

duração  $d[i]$  da tarefa  $i$  ( $i = 1, \dots, t$ )

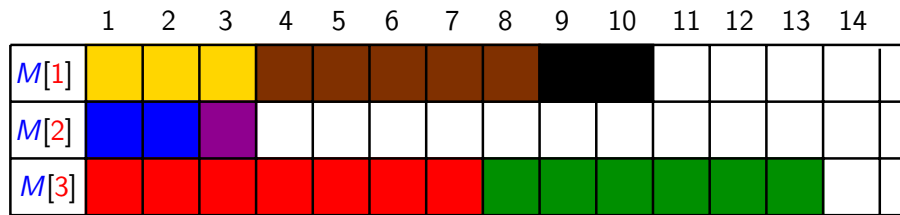
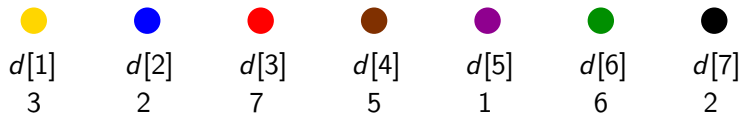
um escalonamento é

uma partição  $\{M[1], \dots, M[m]\}$  de  $\{1, \dots, t\}$



# Exemplo 1

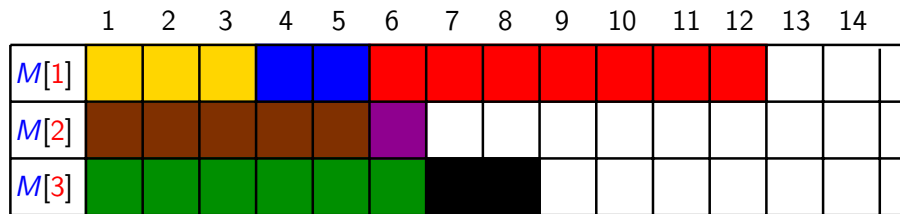
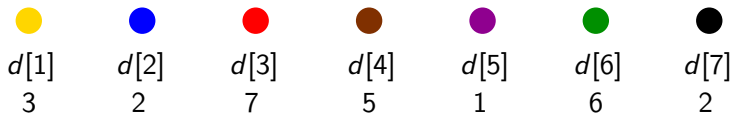
$$m = 3 \quad t = 7$$



$\{\{1, 4, 7\}, \{2, 5\}, \{3, 6\}\} \Rightarrow$  Tempo de conclusão = 13

## Exemplo 2

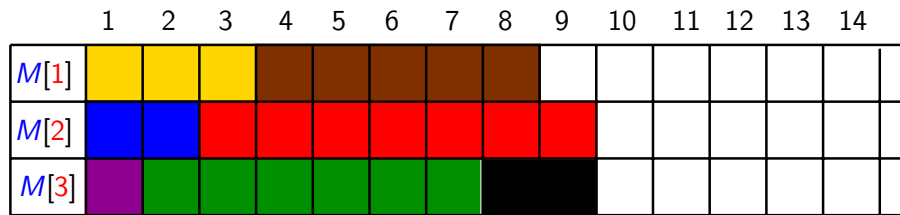
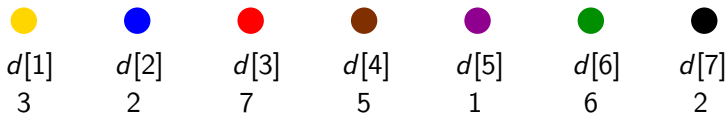
$$m = 3 \quad t = 7$$



$\{\{1, 2, 3\}, \{4, 5\}, \{6, 7\}\} \Rightarrow$  Tempo de conclusão = 12

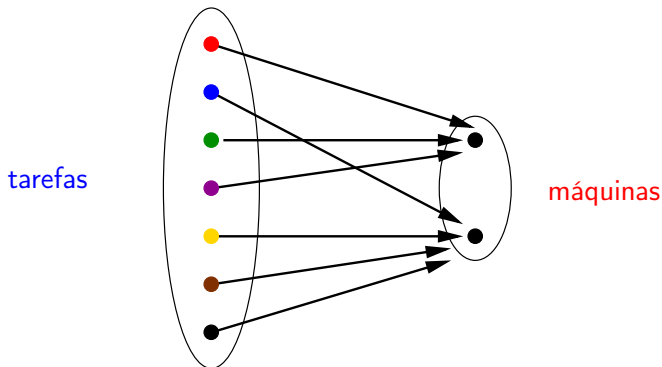
# Problema

Encontrar um escalonamento com tempo de conclusão **mínimo**.



$\{\{1, 4\}, \{2, 3\}, \{5, 6, 7\}\} \Rightarrow$  Tempo de conclusão = 9

## NP-difícil mesmo para $m = 2$



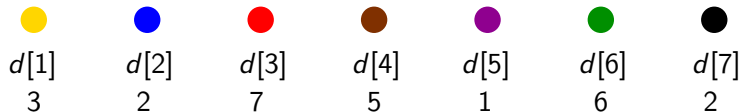
**Algoritmo:** testa todo  $M[1] \subseteq \{1, \dots, t\}$  e escolhe melhor  
 $2^t$  subconjuntos  $\Rightarrow$  **exponencial**

**NP-difícil**  $\Rightarrow$  é improvável que exista algoritmo **polinomial**  
que resolva o problema (se existir, **P = NP**)



# Algoritmo de Graham

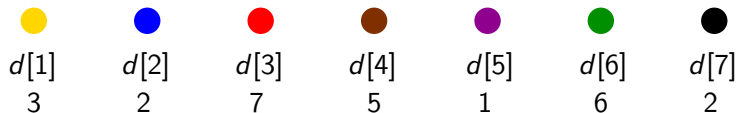
Atribui, uma a uma, cada tarefa à máquina menos ocupada.






	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$M[1]$														
$M[2]$														
$M[3]$														

# Algoritmo de Graham

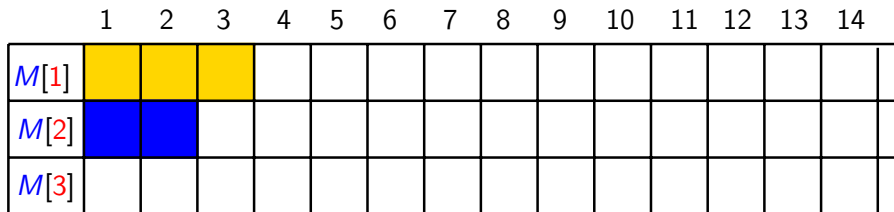
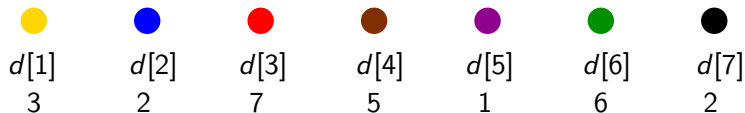
Atribui, uma a uma, cada tarefa à máquina menos ocupada.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$M[1]$														
$M[2]$														
$M[3]$														

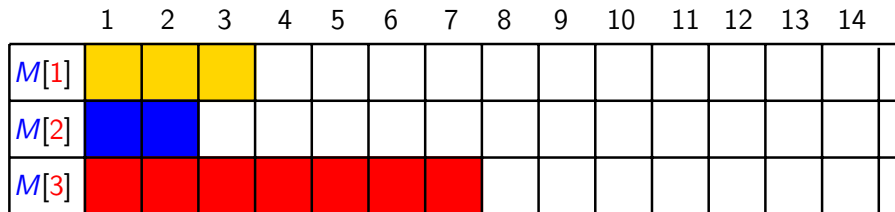
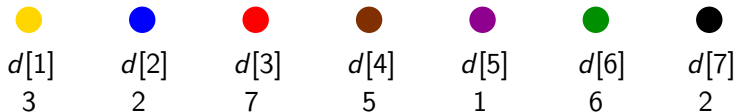
# Algoritmo de Graham

Atribui, uma a uma, cada tarefa à máquina menos ocupada.










# Algoritmo de Graham


















Atribui, uma a uma, cada tarefa à **máquina menos** ocupada.



# Algoritmo de Graham

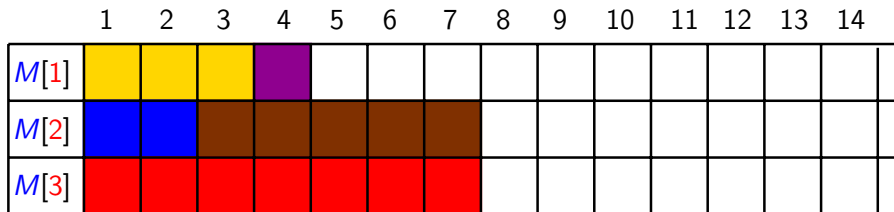
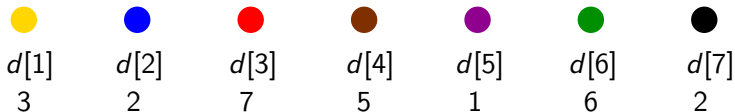
Atribui, uma a uma, cada tarefa à máquina menos ocupada.

						
$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$	$d[6]$	$d[7]$
3	2	7	5	1	6	2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$M[1]$														
$M[2]$														
$M[3]$														

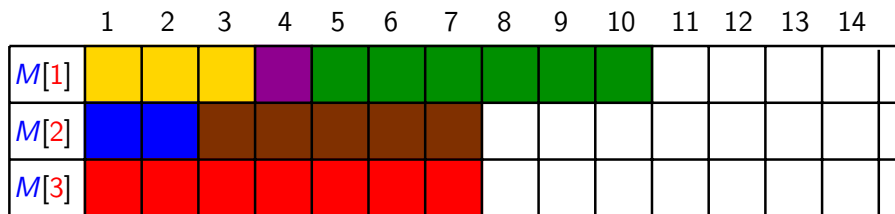
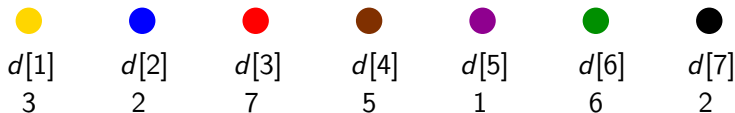
# Algoritmo de Graham

Atribui, uma a uma, cada tarefa à máquina menos ocupada.



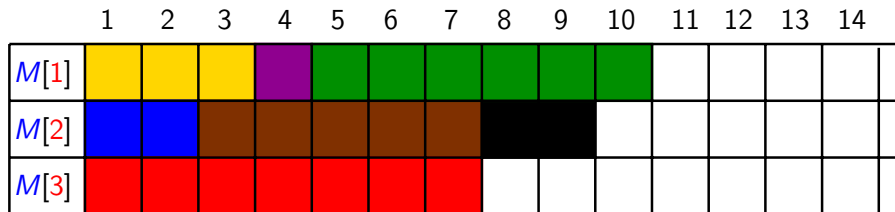
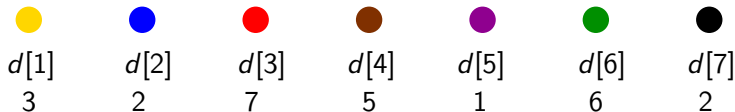
# Algoritmo de Graham

Atribui, uma a uma, cada tarefa à máquina menos ocupada.



# Algoritmo de Graham

Atribui, uma a uma, cada tarefa à máquina menos ocupada.





## Delimitações para OPT

OPT = menor tempo de conclusão de um escalonamento

# Delimitações para OPT

OPT = menor tempo de conclusão de um escalonamento

- ▶ Duração da tarefa mais longa:

$$\text{OPT} \geq \max\{d[1], d[2], \dots, d[t]\}$$

# Delimitações para OPT

OPT = menor tempo de conclusão de um escalonamento

- ▶ Duração da tarefa mais longa:

$$\text{OPT} \geq \max\{d[1], d[2], \dots, d[t]\}$$

- ▶ Distribuição balanceada:

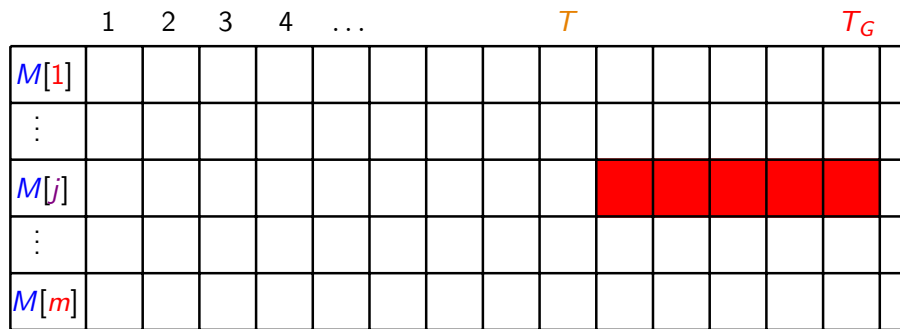
$$\text{OPT} \geq \frac{d[1] + d[2] + \dots + d[t]}{m}$$

## Qualidade do escalonamento de Graham

$T_G$ : conclusão do escalonamento obtido pelo algoritmo

tarefa  $i$ : tarefa que termina de executar no instante  $T_G$

instante  $T$ : instante anterior ao início da execução da tarefa  $i$



## Qualidade do escalonamento de Graham

$T_G$ : conclusão do escalonamento obtido pelo algoritmo

tarefa  $i$ : tarefa que termina de executar no instante  $T_G$

instante  $T$ : instante anterior ao início da execução da tarefa  $i$

	1	2	3	4	...											
$M[1]$																
$\vdots$																
$M[j]$																
$\vdots$																
$M[m]$																

Detailed description of the Gantt chart: The chart shows the execution of  $m$  tasks,  $M[1]$  through  $M[m]$ , over time. The horizontal axis represents time, with discrete steps labeled 1, 2, 3, 4, ...,  $T$ , ...,  $T_G$ . Each task's execution is represented by a horizontal bar. Tasks  $M[1]$  through  $M[m]$  are shown in cyan. Task  $M[j]$  is shown in red. The cyan bars for all tasks end at time  $T$ . The red bar for task  $M[j]$  starts at time  $T$  and ends at time  $T_G$ . The remaining space between  $T$  and  $T_G$  is filled with cyan bars, indicating that other tasks are also executing during this interval.

$$T \cdot m < d[1] + \dots + d[t] \Rightarrow T < \frac{d[1] + \dots + d[t]}{m}$$

## Qualidade do escalonamento de Graham

	1	2	3	4	...										
$M[1]$															
$\vdots$															
$M[j]$															
$\vdots$															
$M[m]$															

$$\begin{aligned}T_G &= T + d[i] \\ &< \frac{d[1] + \dots + d[t]}{m} + d[t] \\ &\leq \frac{d[1] + \dots + d[t]}{m} + \max\{d[1], \dots, d[t]\} \\ &\leq \text{OPT} + \text{OPT} = 2 \text{OPT}\end{aligned}$$

# Conclusão

**Recebe** números inteiros positivos  $m$  e  $t$  e um vetor  $d[1..t]$  e **devolve** um escalonamento de  $\{1, \dots, t\}$  em  $m$  máquinas.

ESCALONAMENTO-GRAHAM ( $m, t, d$ )

- 1 **para**  $j \leftarrow 1$  até  $m$  faça
- 2      $M[j] \leftarrow \emptyset$
- 3      $T[j] \leftarrow 0$
- 4 **para**  $i \leftarrow 1$  até  $t$  faça
- 5     seja  $k$  tal que  $T[k]$  é mínimo
- 6      $M[k] \leftarrow M[k] \cup \{i\}$
- 7      $T[k] \leftarrow T[k] + d[i]$
- 8 **devolva**  $\{M[1], \dots, M[m]\}$

O algoritmo ESCALONAMENTO-GRAHAM é uma 2-aproximação.

# MAC0328 Algoritmos em Grafos

**Objetivos:** Estudar algoritmos para problemas fundamentais em grafos.



# MAC0328 Algoritmos em Grafos

**Objetivos:** Estudar algoritmos para problemas fundamentais em grafos.

**Programa:**

- ▶ Conexão de grafos: componentes, grafos biconexos.
- ▶ Digrafos fortemente conexos (alg. Kosaraju-Sharir, alg. Tarjan).
- ▶ Emparelhamentos máximos em grafos bipartidos.
- ▶ Emparelhamentos em grafos arbitrários (alg. Edmonds).
- ▶ Fluxo máximo (alg. Ford-Fulkerson).
- ▶ Coloração de vértices.
- ▶ Circuitos hamiltonianos.
- ▶ Tópicos opcionais:  
link analysis, network analysis, redes aleatórias.

# MAC0328 Algoritmos em Grafos

**Objetivos:** Estudar algoritmos para problemas fundamentais em grafos.

**Programa:**

- ▶ Conexão de grafos: componentes, grafos biconexos.
- ▶ Digrafos fortemente conexos (alg. Kosaraju-Sharir, alg. Tarjan).
- ▶ Emparelhamentos máximos em grafos bipartidos.
- ▶ Emparelhamentos em grafos arbitrários (alg. Edmonds).
- ▶ Fluxo máximo (alg. Ford-Fulkerson).
- ▶ Coloração de vértices.
- ▶ Circuitos hamiltonianos.
- ▶ Tópicos opcionais:  
link analysis, network analysis, redes aleatórias.

**Bibliografia principal:**

Sedgewick & Wayne, Algorithms, 4a ed., 2011.

# MAC0414 Autômatos, Computabilidade e Complexidade

**Objetivos:** Estudo de vários formalismos para computação e algoritmos e as limitações de certas formas de computação.

# MAC0414 Autômatos, Computabilidade e Complexidade

**Objetivos:** Estudo de vários formalismos para computação e algoritmos e as limitações de certas formas de computação.

**Programa:**

- ▶ Palavras, linguagens, operações sobre linguagens.
- ▶ Linguagens regulares.
- ▶ Autômatos finitos determinísticos e não determinísticos.
- ▶ Teorema de Kleene.
- ▶ Autômatos reduzidos.
- ▶ Modelos de computação; máquinas de Turing.
- ▶ Tese de Church.
- ▶ Redutibilidade e problemas indecidíveis.
- ▶ Complexidade, problemas decidíveis em tempo polinomial.
- ▶ Não-determinismo versus determinismo.
- ▶ Redutibilidade e problemas NP-completos.
- ▶ Teorema de Cook-Levin.

# MAC0414 Autômatos, Computabilidade e Complexidade

**Objetivos:** Estudo de vários formalismos para computação e algoritmos e as limitações de certas formas de computação.

**Programa:**

- ▶ Palavras, linguagens, operações sobre linguagens.
- ▶ Linguagens regulares.
- ▶ Autômatos finitos determinísticos e não determinísticos.
- ▶ Teorema de Kleene.
- ▶ Autômatos reduzidos.
- ▶ Modelos de computação; máquinas de Turing.
- ▶ Tese de Church.
- ▶ Redutibilidade e problemas indecidíveis.
- ▶ Complexidade, problemas decidíveis em tempo polinomial.
- ▶ Não-determinismo versus determinismo.
- ▶ Redutibilidade e problemas NP-completos.
- ▶ Teorema de Cook-Levin.

**Bibliografia principal:**

Sipser, Introduction to the Theory of Computation, 3a ed., 2012.

# Módulo Algoritmos

## Demais disciplinas:

- ▶ MAC0325 Otimização Combinatória
- ▶ MAC0327 Desafios de Programação I
- ▶ MAC0345 Desafios de Programação II
- ▶ MAC0385 Estruturas de Dados Avançadas
- ▶ MAC0333 Armazenamento e Recuperação de Informação
- ▶ MAC0331 Geometria Computacional
- ▶ MAC0336 Criptografia para Segurança de Dados
- ▶ MAC0450 Algoritmos de Aproximação
- ▶ MAC0465 Biologia Computacional
- ▶ MAC0466 Teoria dos Jogos Algorítmica

Vejam fazer a trilha de teoria!!!!

Obrigada!

Perguntas???

