

# Algoritmos Genéticos: Conceitos e Aplicações

**Prof. Dr. Peterson Belan**  
**belan@uni9.pro.br**

# Algoritmos Genéticos (AG)

Os algoritmos genéticos foram introduzidos por John Holland em 1975.

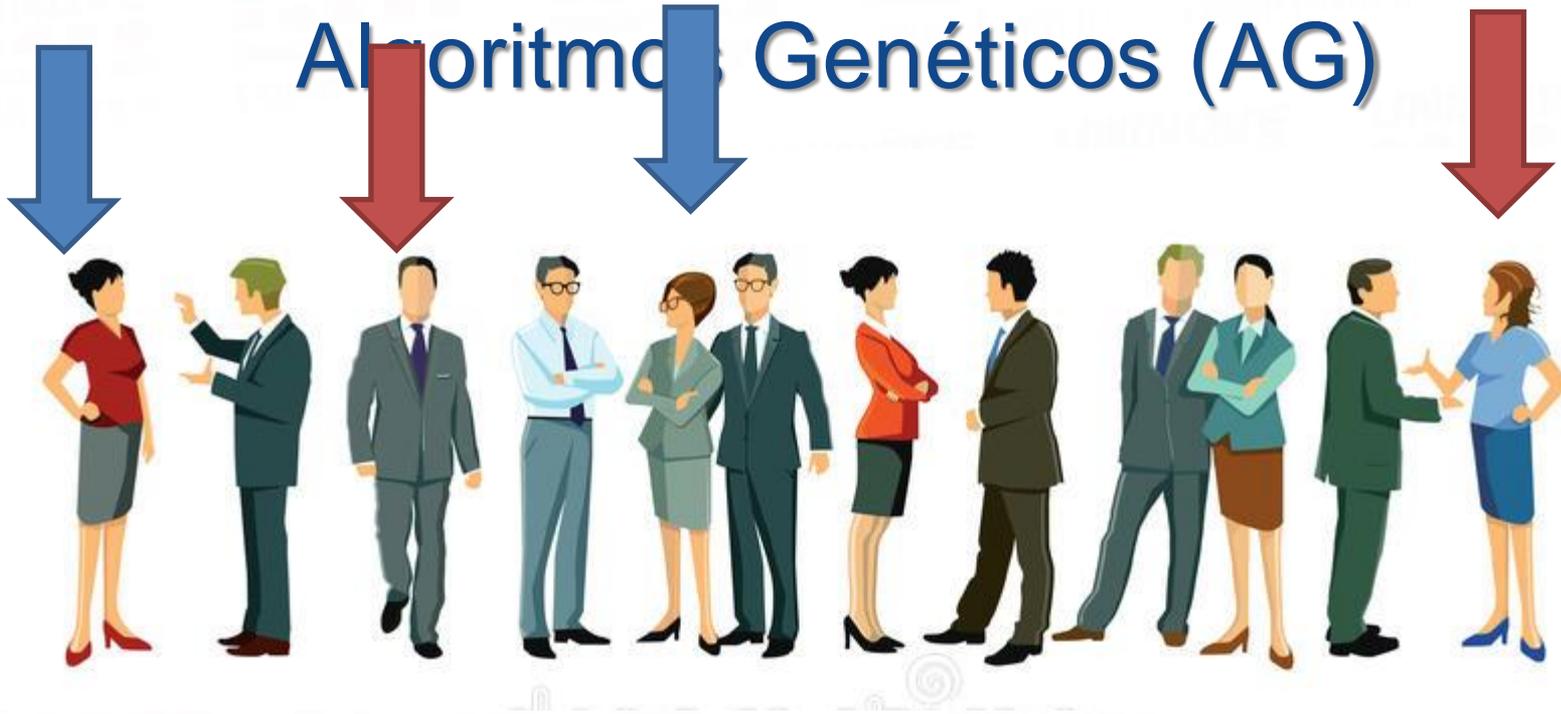
É um método de busca e otimização baseado no processo de seleção natural que simula a evolução das espécies (GOLDBERG, 1989).

**AG** é um dos algoritmos meta-heurísticos pertinentes ao ramo da **computação evolutiva (CE)** voltado para **resolução de problemas de otimização** via populações.

# Algoritmos Genéticos (AG)



# Algoritmos Genéticos (AG)



# Algoritmos Genéticos (AG)



dreamstime.



# Algoritmos Genéticos (AG)



dreamstime.



# Algoritmos Genéticos (AG)



dreamstime

Me  
adap  
ação

Me  
adap  
ação

# Algoritmos Genéticos (AG)

Possível  
solução

Possível  
solução

Possível  
solução



Função  
objetivo

# Algoritmos Genéticos (AG)

Geração  
1



# Algoritmos Genéticos (AG)

Geração 1



Geração 2



# Algoritmos Genéticos (AG)

Geração 1



Geração 2



Geração 3



# Algoritmos Genéticos (AG)



# Algoritmos Genéticos (AG)

Intensificação

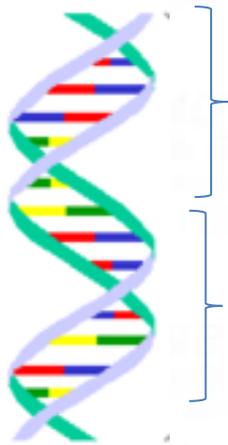
diversificação

reprodução

mutação



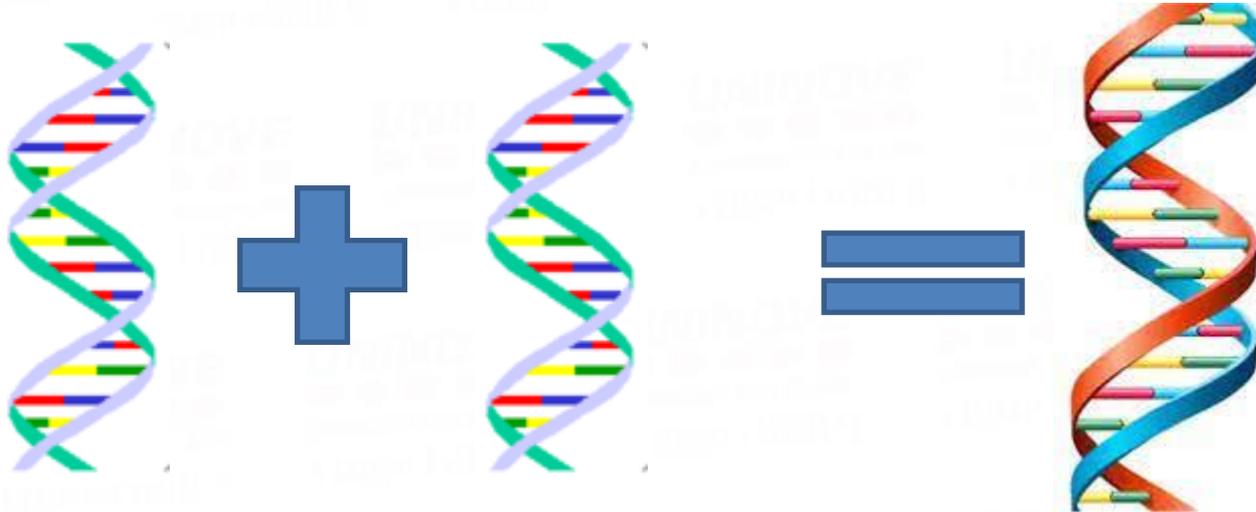
# Algoritmos Genéticos (AG)



Gene 1

Gene 2

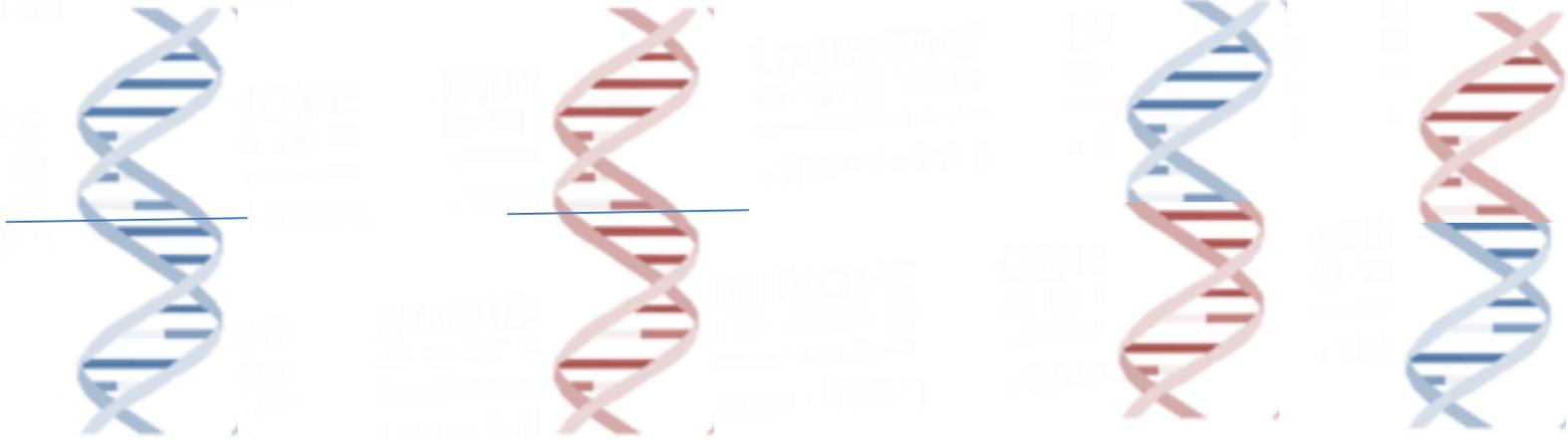
# Algoritmos Genéticos (AG)



Cromossomos parentais

Cromossomo descendente

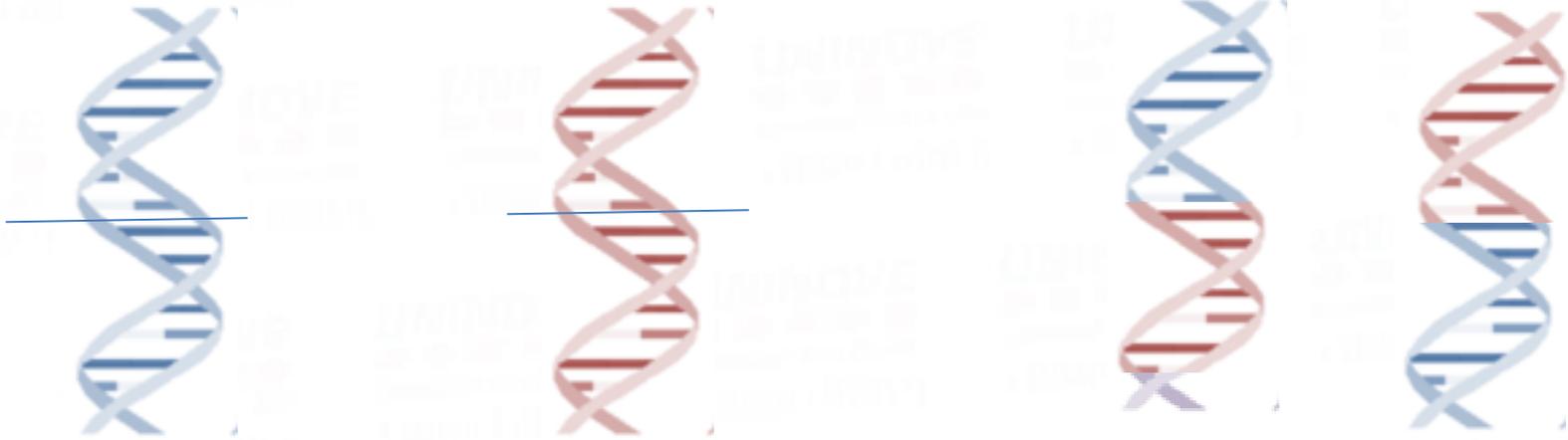
# Algoritmos Genéticos (AG)



Corta-se o cromossomo e recombina

reprodução

# Algoritmos Genéticos (AG)



Corta-se o cromossomo e  
recombina

Mutação:  
alteração do  
alelo

# Algoritmos Genéticos (AG)

- Taxa de reprodução
- Probabilidade de mutação
- Controle populacional
- Critério de parada
- Função objetivo
- Restrições

# Otimização

Otimizar consiste em encontrar valores mínimos ou máximos de uma função, denominada **Função Objetivo**, que pode envolver uma ou mais variáveis, normalmente com valores em intervalos que delimitam uma região do espaço de busca multi-dimensional.

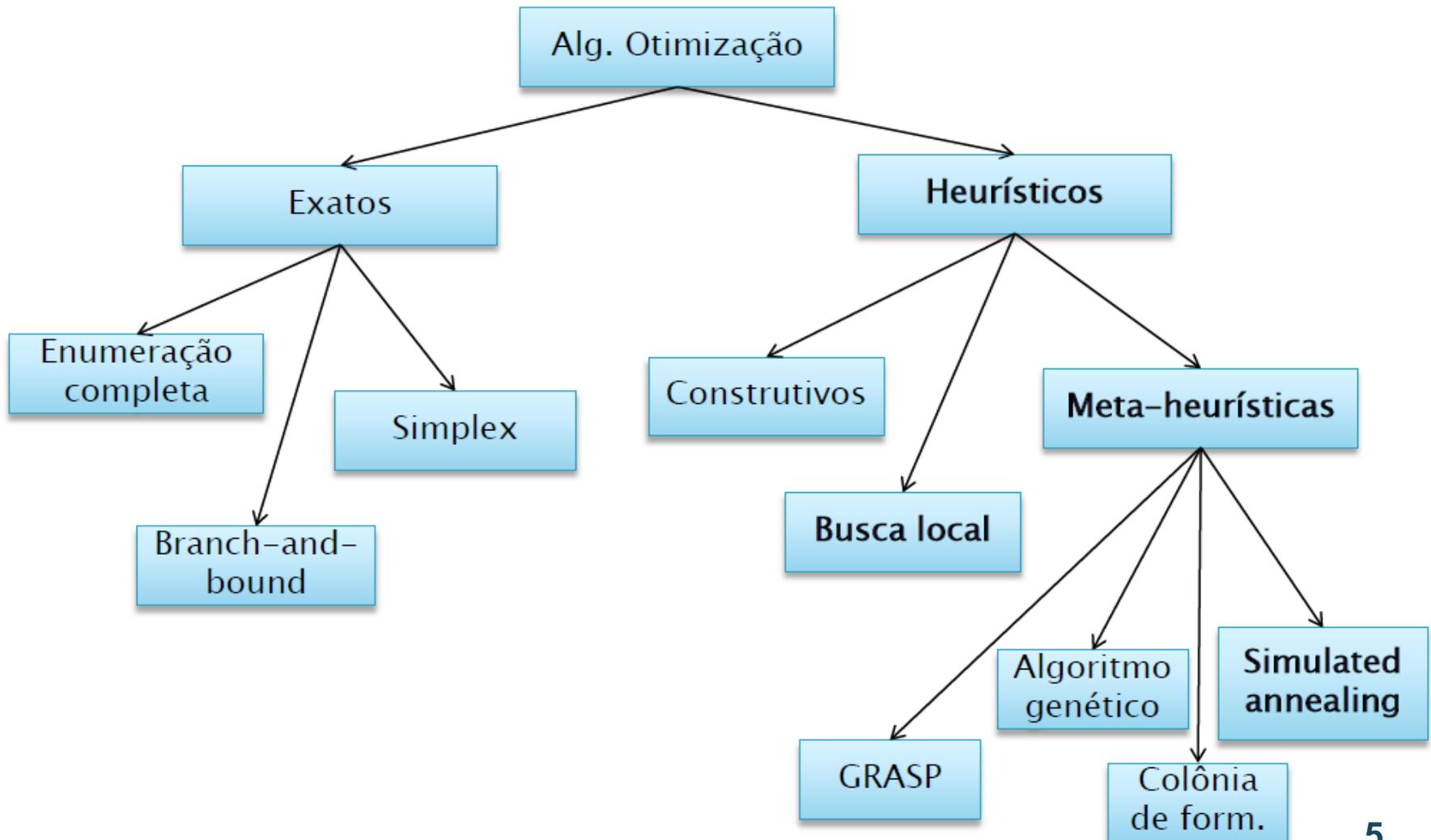
## **De forma bastante resumida:**

*“é o processo de tentar encontrar a melhor solução dentre todas as possíveis para um determinado problema”.*

# Onde os probl. de otimização aparecem?

- Roteamento (veículos, aeronaves, robôs, etc.)
- Alocação de horários/salas
- Gerenciamento de projetos
- Processos produtivos (sequenciamento da produção - Job Shop, corte e empacotamento, etc)
- Estimação de parâmetros em processos industriais
- Portfólio de ações
- Datamining

# Métodos de Resolução de Problemas de Otimização



# Algoritmos Genéticos (AG)

Os AGs geram os indivíduos (cadeias de bits) também chamados de cromossomo, para evoluírem em busca da solução de um dado problema. Cada indivíduo representa uma possível solução para o problema e é analisado com base em uma função de aptidão (Função Objetivo).

Por meio de uma “competição”, os indivíduos mais aptos são selecionados e, em seguida, cruzados para gerarem novos indivíduos mais aptos e melhores que os anteriores. Assim, a cada geração há uma maior probabilidade de um dos indivíduos estar próximo de uma solução para o problema.

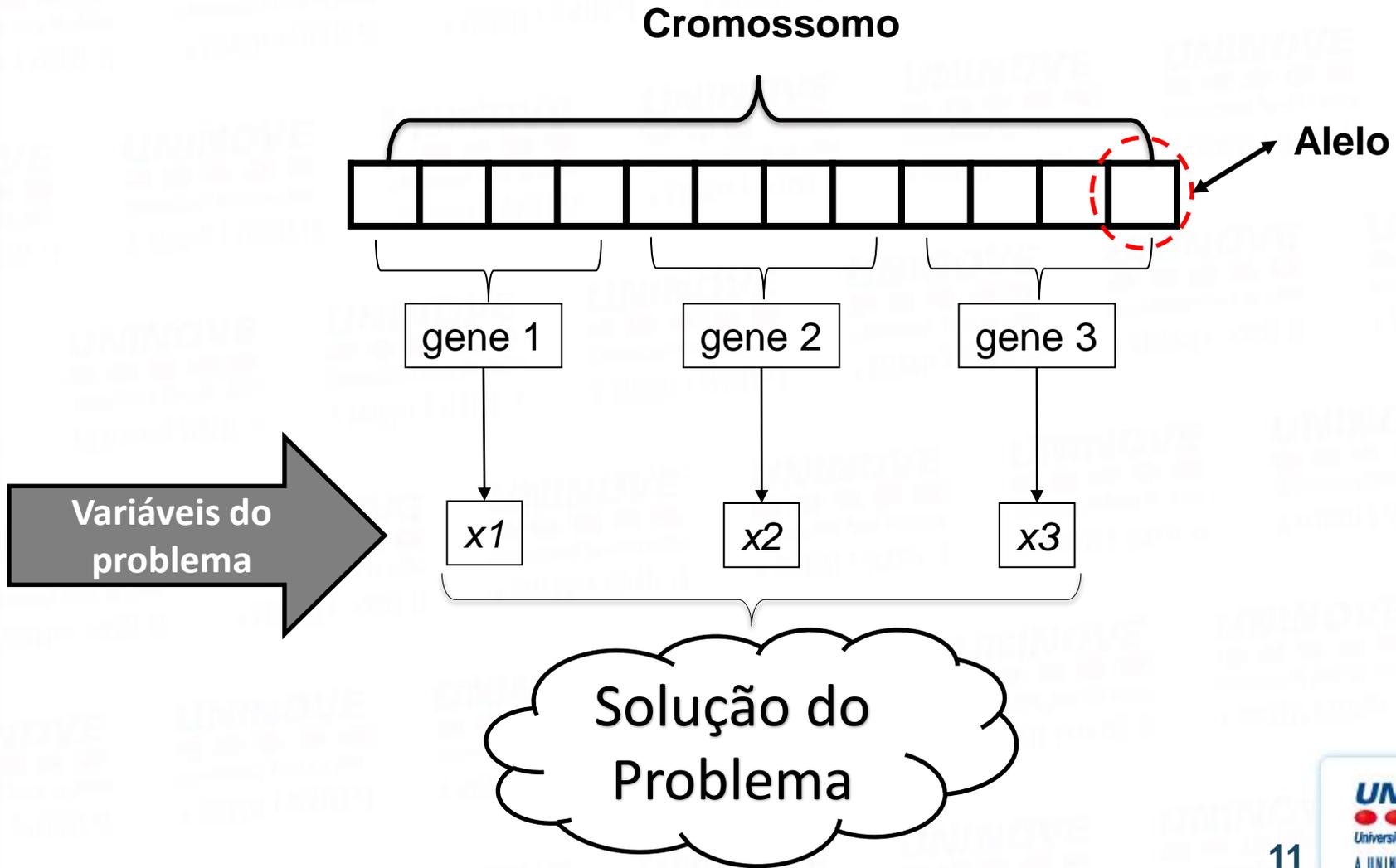
# Algumas características do AG

- AG manipula uma população de indivíduos.
- Indivíduos são possíveis soluções do problema.
- Os indivíduos são combinados (*crossover*) uns com os outros, produzindo filhos que podem ou não sofrer mutação.
- As populações evoluem através de sucessivas gerações até encontrar a solução ótima ou atingir um critério de parada.

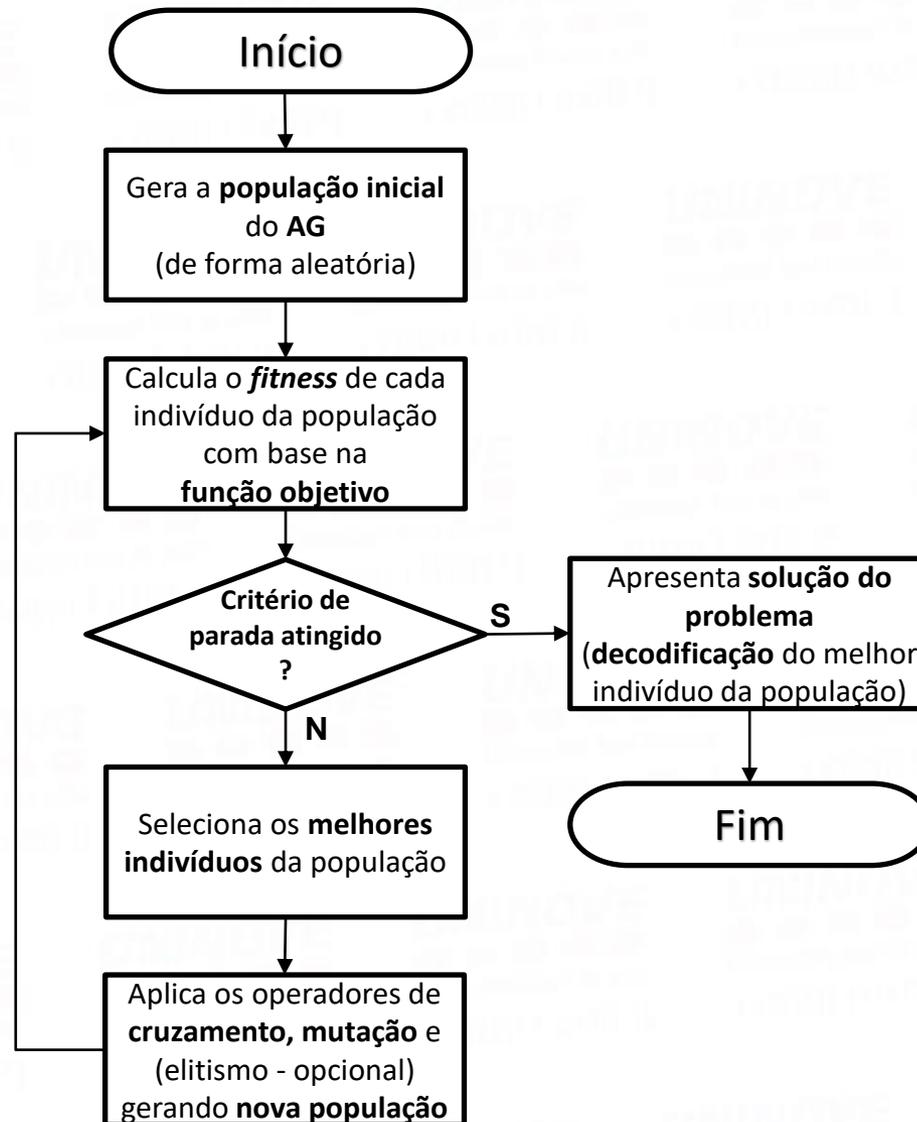
# Elementos do AG

<b>AG</b>	<b>Problema de Otimização</b>
Cromossomo	Representação de uma solução: string, número binário, número real, vetor, etc
Indivíduo	Solução de um problema
População	Conjunto de soluções
Gene	Parte da representação de uma solução; normalmente uma variável
Alelo	Valores que um gene pode assumir

# Estrutura de um Cromossomo



# Fluxograma do funcionamento do AG



# Cruzamento e Mutação

Pai 1 = 10001|10101

Pai 2 = 11101|11011

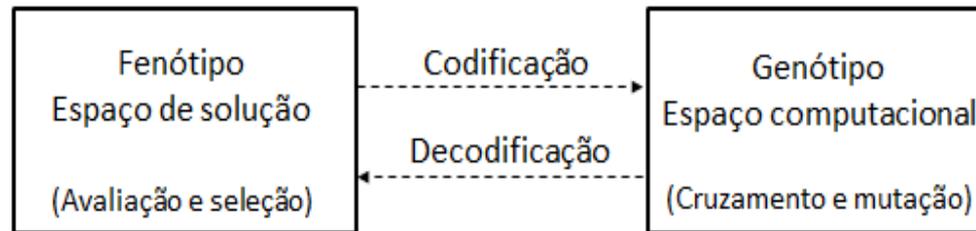
Filho 1 = 1000111011

Filho 2 = 1110110101  mutação → 1110111101

O **cruzamento** é uma operação essencial para que os filhos herdem características genéticas dos seus pais, propiciando a evolução de uma geração para outra.

A **mutação** é responsável por criar diversidade populacional, a partir da mudança de uma ou mais características (alelos) de um cromossomo.

# Codificação / Decodificação do Cromossomo



- **Genótipo:** está relacionado à população no espaço computacional, no qual as soluções são representadas para serem facilmente entendidas e manipuladas computacionalmente;
- **Fenótipo:** está relacionado com a população no espaço de soluções do mundo real, é a estrutura construída a partir das informações do genótipo que permite a decodificação do cromossomo;
- **Codificação e Decodificação:** codificação é uma operação que transforma uma solução do espaço fenótipo para o espaço genotípico, enquanto a decodificação é responsável pela transformação inversa, como mostra a Figura acima.

# Codificação / Decodificação do Cromossomo

A **Decodificação** do cromossomo depende do tipo de cada variável de decisão que compõe a solução do problema de otimização a ser resolvido, o qual deve estar formalizado, por meio de uma **função objetivo (FO)**, da seguinte maneira:

**Minimizar / Maximizar função objetivo (variáveis de decisão)**

**Sujeito a:**

**(lista de restrições - equações ou inequações)**

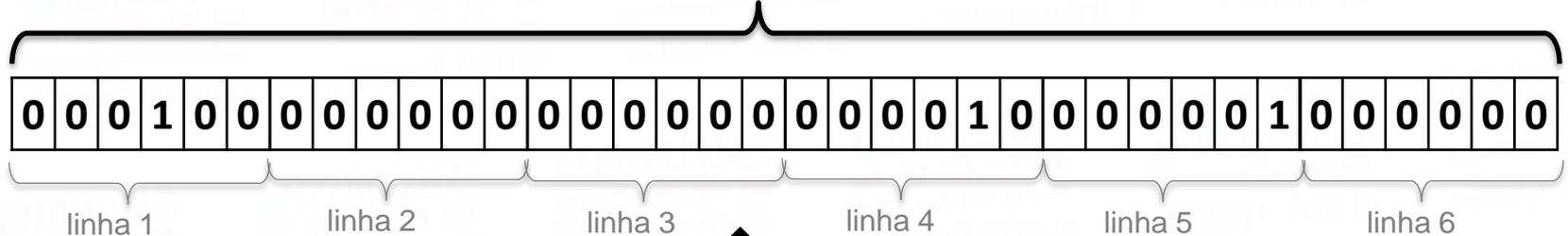
**(tipos das variáveis de decisão e limites inferiores e superiores explicitados)**

As variáveis de decisão de uma **FO** podem ou não ser do mesmo tipo (binário, inteiro ou real). Por este motivo, a decodificação dos genes pode ser feita por meio de procedimentos diferentes. O procedimento para decodificação de cada um dos tipos é mostrado a seguir.

# Decodificação do Cromossomo

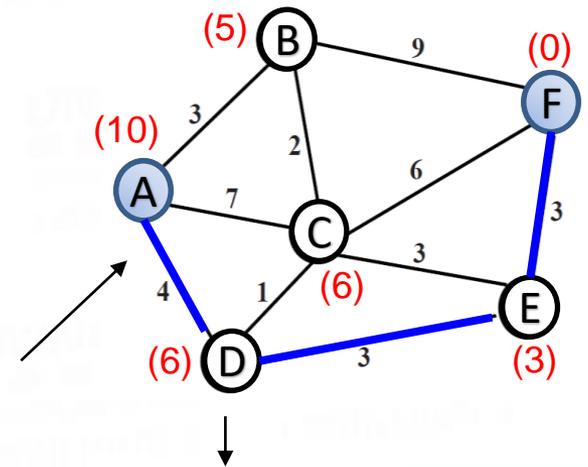
Se as variáveis de decisão (que solucionam o problema) **são números binários**, em geral, a decodificação acaba sendo mais simples. No exemplo abaixo é feito apenas um **rearranjo do cromossomo**, sendo uma decodificação mais direta.

Cromossomo



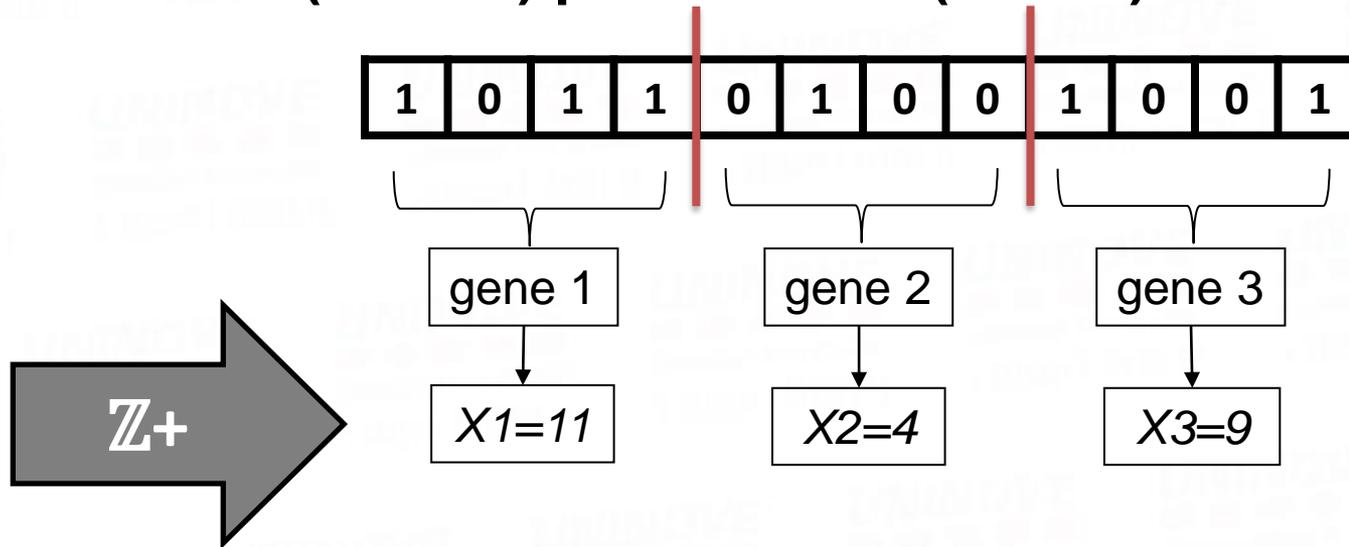
Solução do Problema

	A	B	C	D	E	F
A	0	0	0	1	0	0
B	0	0	0	0	0	0
C	0	0	0	0	0	0
D	0	0	0	0	1	0
E	0	0	0	0	0	1
F	0	0	0	0	0	0



# Decodificação do Cromossomo

Se as variáveis de decisão são números **inteiros não negativos** ( $\mathbb{Z}^+$ ), então a decodificação consiste na **conversão dos genes de base 2 (binário) para base 10 (inteiro)**.



Relembrando...  
Conversão de base  
2 (binário) para  
base 10



Exemplo1

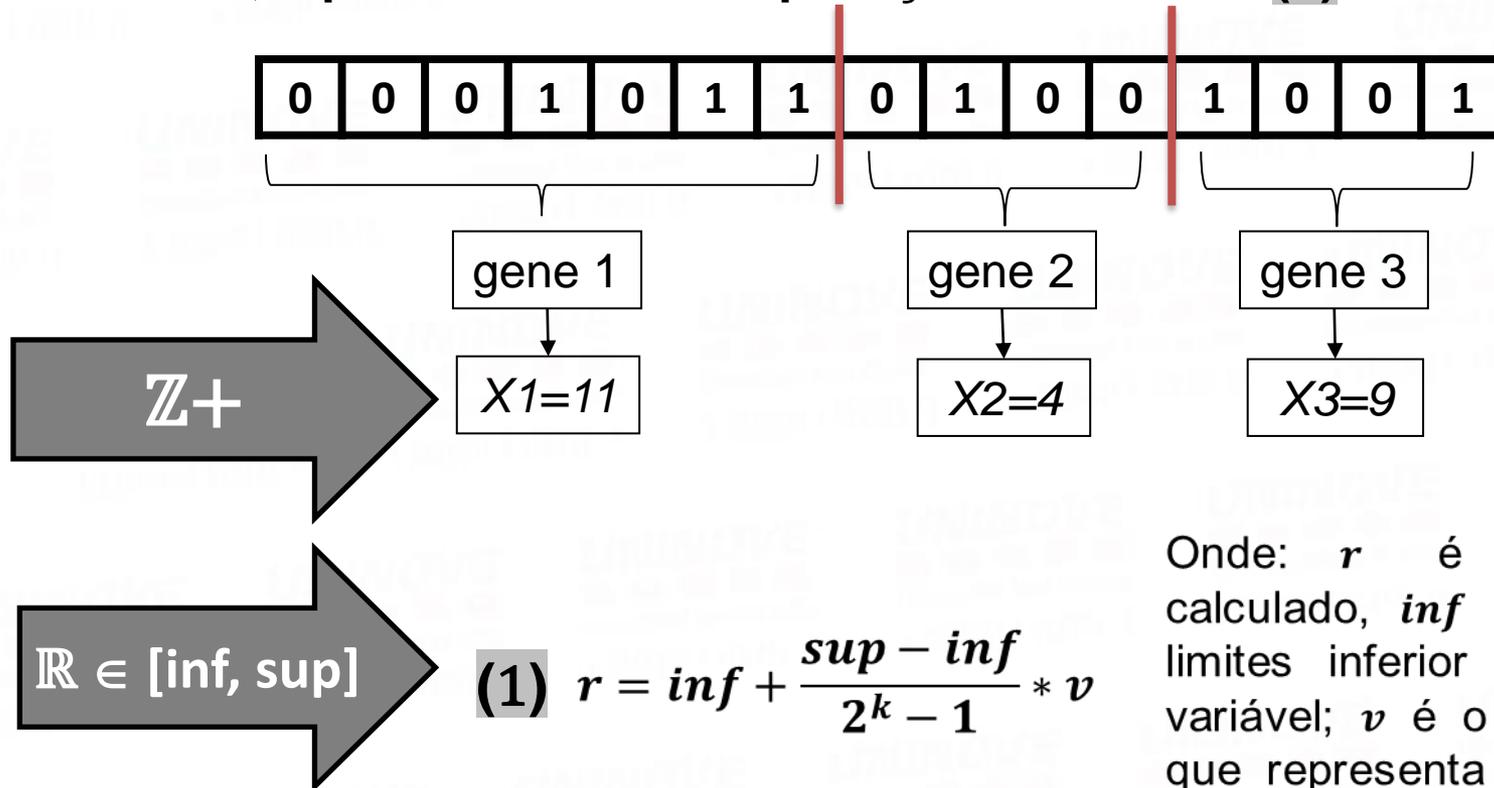
$$\begin{array}{r} 1 \ 0 \ 1 \ 1_2 \\ \left. \begin{array}{l} \rightarrow 1 \times 2^0 = 1 \\ \rightarrow 1 \times 2^1 = 2 \\ \rightarrow 0 \times 2^2 = 0 \\ \rightarrow 1 \times 2^3 = 8 \end{array} \right\} = 11_{10} \end{array}$$

Exemplo2

$$\begin{array}{r} 11001 \\ \left. \begin{array}{l} \rightarrow 1 \times 2^0 = 1 \\ \rightarrow 0 \times 2^1 = 0 \\ \rightarrow 0 \times 2^2 = 0 \\ \rightarrow 1 \times 2^3 = 8 \\ \rightarrow 1 \times 2^4 = 16 \end{array} \right\} = 25_{10} \end{array}$$

# Decodificação do Cromossomo

Se as variáveis de decisão são números reais, ou mesmo inteiros ( $\mathbb{R}$  ou  $\mathbb{Z}$ ), então a decodificação consiste na **conversão dos genes de base 2 para base 10**, e posteriormente a aplicação da fórmula (1) abaixo.



Ex. gene = 0001011,  $inf = -5,0$  e  $sup = 5,0$ ,

então:  $r = -5 + \frac{5 - (-5)}{2^7 - 1} * 11 = -4,133$

Onde:  $r$  é o valor real calculado,  $inf$  e  $sup$  são os limites inferior e superior da variável;  $v$  é o valor do gene que representa a variável (na base 10) e  $k$  é o comprimento do gene.

# Mas, como calcular o comprimento de cada gene?

O comprimento de cada gene (número de bits necessários para representar uma determinada informação) é calculado com base na fórmula:

$$2^k > (\mathbf{sup} - \mathbf{inf}) * 10^p \quad (2)$$

Na qual:

$k$  : comprimento do gene

$\mathbf{sup}$  : limite superior da variável

$\mathbf{inf}$  : limite inferior da variável

$p$  : precisão da variável (número de casas depois da vírgula)

Essa fórmula é sempre utilizada, independentemente do tipo de dado que será codificado no cromossomo.

# Mas, como calcular o comprimento de cada gene?

Exemplo: Considere uma variável cujos limites são:  $\text{inf} = -5,0$  e  $\text{sup} = 5,0$ :

Neste caso,  $p = 1$  (como visto, há uma casa depois da virgula em ambos os números que representam os limites inferior e superior da variável)

$$2^k > (\text{sup} - \text{inf}) * 10^p$$

$$2^k > (5 - -(-5)) * 10^1$$

$$2^k > 10 * 10^1$$

$$2^k > 100$$

Logo  $k=7$ , ou seja, o menor valor inteiro possível que  $k$  pode assumir para que a expressão  $2^k > 100$  seja verdadeira. O valor de  $k$  poderia ser obtido assim:  $\log_2(100) \cong 6,644$ . Como não podemos usar 6,644 bits, então adotamos o primeiro valor inteiro acima desse número, ou seja, 7.



# Aplicação de Algoritmos Genéticos (AG)

## Roteiro Básico

Antes de qualquer coisa, deve-se ter em mente que o **AG** é uma técnica **para resolução de problemas de otimização**.

Assim, se o problema que você pretende resolver **não é** ou **não pode** ser formulado como um **problema de otimização**, então o **AG não é a técnica de IA adequada**.

Há muitas outras técnicas de IA para resolver os mais variados tipos de problemas...

# Aplicação de Algoritmos Genéticos (AG)

## Roteiro Básico

Antes de iniciar a aplicação do AG, a primeira pergunta a ser respondida é:

1. O problema está corretamente formalizado como um problema de otimização?

**Minimizar / Maximizar função objetivo (variáveis de decisão)**

· **Sujeito a:**

**(lista de restrições - equações ou inequações)**

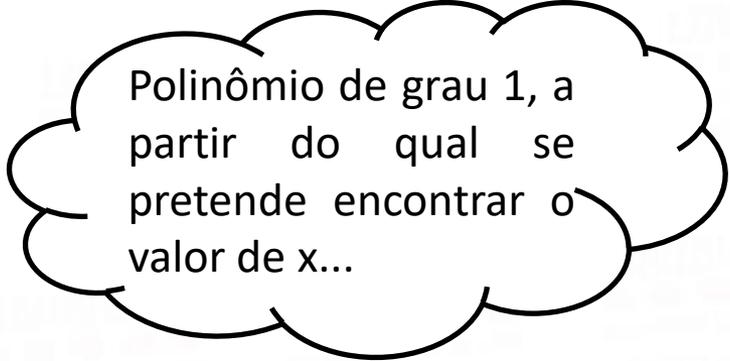
**(tipos das variáveis de decisão e limites inferiores e superiores explicitados)**

Se o problema não estiver formalizado como indicado acima, o primeiro passo é formalizá-lo. **Lembre-se que essa função objetivo (FO) será usada tanto para modelagem do cromossomo quanto para avaliar as soluções produzidas pelo AG (cálculo do *fitness* mencionado no fluxograma de Funcionamento do AG – slide 9).**

# Exemplo

Desenvolvimento de um AG para solucionar o seguinte problema<sup>1</sup>:

$$2x+10=18$$



Polinômio de grau 1, a partir do qual se pretende encontrar o valor de x...

Como sabemos isso **não é naturalmente um problema de otimização**, mas um polinômio de primeiro grau no qual se pretende encontrar o valor da variável x que satisfaça a igualdade.

Contudo, é **possível formalizá-lo como um problema de otimização**, como mostrado a seguir. Essa transformação consiste no primeiro primeiro passo.

*1. Extraído da revista Revista PC Master Edição 64*

# Passo 1

**Formalização do polinômio  $2x+10=18$  como um problema de otimização:**

$$\begin{aligned} &\text{Minimizar } f(x) = |2x+10-18| \\ &\text{sujeito a } x \in \mathbb{Z}^+, 0 \leq x \leq 15 \end{aligned}$$

Note que agora temos um problema de otimização corretamente formalizado, cuja função objetivo (**FO**) a ser minimizada envolve uma única variável de decisão ( $x$ ) e que os limites **inferior** e **superior** de  $x$  são, respectivamente, 0 e 15 (**inf=0** e **sup=15**).

Apesar desses limites não terem vindo do problema original, eles devem ser fornecidos “**sempre**”. Caso contrário, o espaço de busca seria infinito e não teríamos como modelar o AG.

**Obs.:** a maioria esmagadora dos problemas a serem resolvidos com o uso do AG já está formalizada como problema de otimização.

# Passo 2

## **Modelagem do cromossomo / parametrização do AG:**

Neste passo é que devemos definir como a solução do problema será codificada no cromossomo. Assim teremos as informações de genótipo e fenótipo (ver slide 12).

Para tanto, precisamos definir:

- Quantidade de genes do cromossomo
- Tamanho do cromossomo (com base no comprim. de cada gene)
- Ponto(s) de corte (qtd. e posição)
- Tamanho da população
- Esquema de seleção
- Taxa de mutação

# Passo 2

## -Quantidade de genes do cromossomo:

Como a **FO** do problema a ser resolvido envolve uma única variável ( $x$ ), então o cromossomo só terá um gene (ver slide 11).

## -Tamanho do cromossomo:

Uma vez que o cromossomo será formado por um único gene, então ao definir o comprimento do gene teremos o comprimento (tamanho) do cromossomo. Para isso usaremos a **fórmula 2** (ver slides 17 e 18)

$2^k > (\text{sup-inf}) * 10^p$ , na qual  $k$  é o comprimento do gene e  $p$  é a precisão da variável  $x$ .  
 $2^k > (15-0) * 10^0$  Sendo a variável  $x$  inteira, então o número de casas decimais é ZERO.

$$2^k > 15 * 1$$

$$2^k > 15$$

**Logo,  $k = 4$**  (menor número inteiro que  $k$  pode assumir para que a expressão  $2^k > 15$  seja verdadeira)

**Obs.: Caso o cromossomo tenha mais de um gene, essa conta deve ser efetuada para cada gene.**

# Passo 2

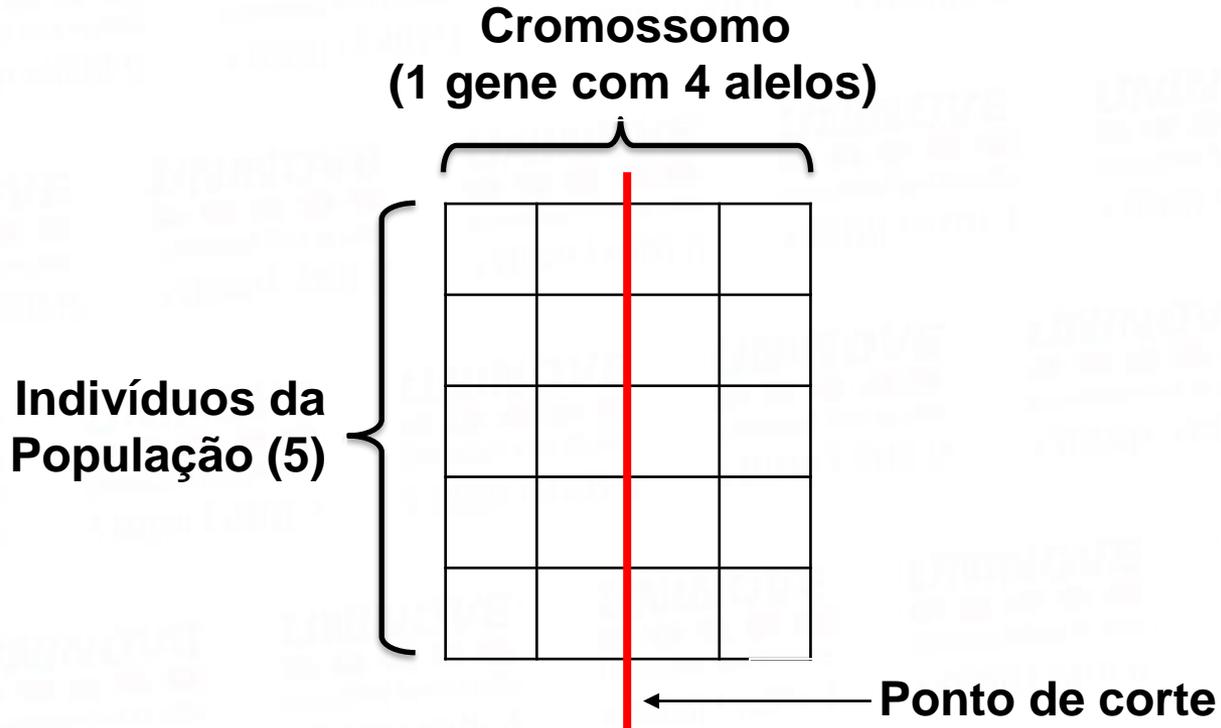
- Ponto(s) de corte (qtd. e posição):** 1 ponto de corte no meio do cromossomo (conforme mostrado no slide 10)
- Tamanho da população:** 5 indivíduos (definido de forma empírica)
- Esquema de seleção:** Ranking com 3 indivíduos selecionados (adotando-se a forma de cruzamento mostrada no slide 7 são necessários 3 indivíduos para recompor a população, pois cada par de indivíduos gera 2 filhos)
- Taxa de mutação:** 10% (definido de forma empírica)

## Observações:

1. Se a taxa de mutação for muito alta, então o AG tende a se tornar aleatório, perdendo as suas principais características;
2. “Empírico” não é sinônimo de aleatório. De acordo com o dicionário, “empírico” significa “baseado na experiência e na observação”

# Passo 2

Com base na parametrização do AG, temos a seguinte configuração:



Como em toda a população há 20 alelos ( $4 \times 5$ ), então a cada geração 2 deles, escolhidos aleatoriamente, serão mutados.

# Passo 3

## Executar o AG:

A execução do AG é feita com base no seu fluxograma de funcionamento (slide 6).

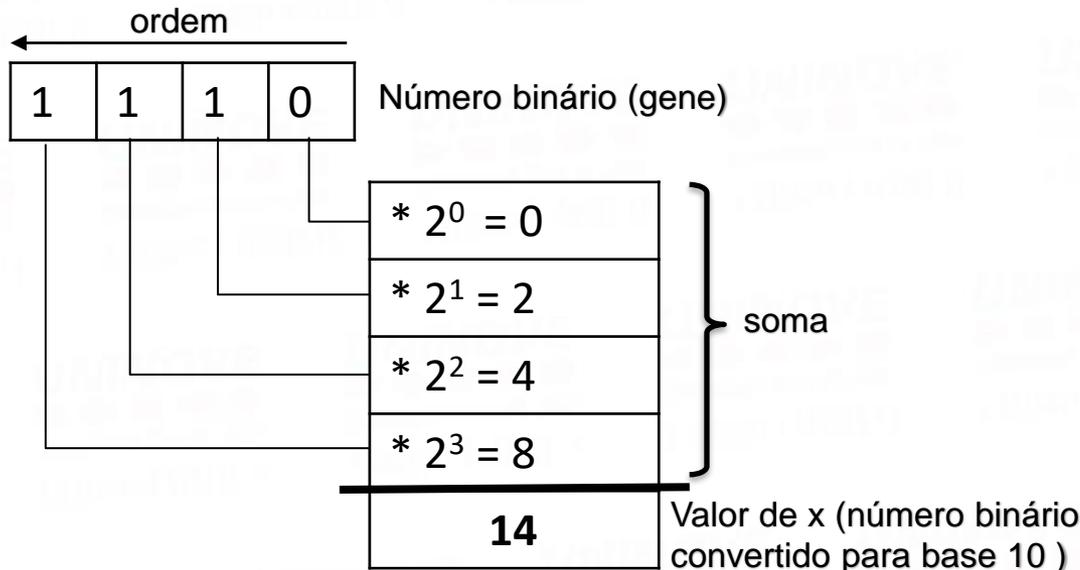
A simulação da execução do AG está na planilha Excel “Planilha\_Simulação\_Funcionamento\_AG\_Problema\_2X.xlsx”, disponível no Google Class.

## Observações:

1. A operação de elitismo é importante, mas não essencial para o AG. Por isso, ela não foi levada em conta na simulação realizada. Elitismo significa, transportar uma fração dos indivíduos da elite (melhores do ranking) de uma geração para a próxima.
2. Como a solução do problema (valor de  $x$ ) consiste em um número inteiro não negativo, então o esquema de decodificação mostrado no slide 15 deve ser empregado para avaliar cada solução gerada pelo AG.

# Passo 3

## Exemplo de avaliação de uma solução (cromossomo) gerada pelo AG



Procedimento para decodificação da solução encapsulada no cromossomo: conversão de base 2 (binário) para base 10 (inteiro não negativo)

$$f(x) = |2 \times 14 + 10 - 18| = 20 \leftarrow \text{Valor da função objetivo (FO) ou fitness do cromossomo}$$

Esse procedimento é repetido para cada indivíduo (cromossomo) da população.

# Exercício 1 - Problema de otimização da produção

Considere uma microempresa que fabrica dois tipos de produto: rádio standard e rádio luxo. Com relação ao rádio standard temos as seguintes informações:

- (i) A linha de produção comporta um máximo de 24 pessoas;
- (ii) Cada rádio consome 1 homem/dia para ser produzido;
- (iii) Cada rádio fornece um lucro de R\$ 30,00.

Com relação ao rádio luxo:

- (i) A linha de produção comporta um máximo de 32 pessoas;
- (ii) Cada rádio consome 2 homens/dia para ser produzido;
- (iii) Cada rádio fornece um lucro de R\$ 40,00.

A fábrica possui 40 empregados a serem alocados nas duas linhas de produção. O objetivo é maximizar o lucro. Quantos rádios de cada tipo (standard e luxo) devem ser produzidos para maximizar o lucro?

Tal problema pode ser representado matematicamente da seguinte maneira:

$$\text{Maximizar } f(x_1, x_2) = 30x_1 + 40x_2$$

$$\text{Sujeito a: } x_1 + 2x_2 \leq 40$$

$$x_1 \leq 24$$

$$x_2 \leq 16$$

$$x_1 \geq 0, x_2 \geq 0,$$

**Tomando como base a função objetivo apresentada, pede-se:**

**1)** Modele um AG para resolver o problema. Isso consiste em definir: quantidade de genes do cromossomo, com seus respectivos comprimentos (qtd. de bits); comprimento do cromossomo (com base nos comprimentos dos genes); ponto(s) de corte; tamanho da população (mínimo 5 indivíduos não nulos e não repetidos); esquema de seleção e qtd. indivíduos selecionados e taxa de mutação (no máximo 10% do total de alelos de toda a população).

**2)** Simule a execução do AG (com pelo menos 3 gerações) e apresente a solução encontrada.

# Exercício 1 - Resolução

**Passo 1. OK** (é probl. de otimização e está corretamente formalizado)

**Passo 2. Modelagem do cromossomo**

-Quantidade de genes do cromossomo: 2 genes (variável  $x_1$ : inf=0, sup=24; variável  $x_2$ : inf=0, sup=16; como são números inteiros, então precisão de 0 casas decimais para ambas as variáveis)

-Tamanho do cromossomo (com base no comprim. de cada gene): como são 2 genes, então precisamos calcular o comprimento de cada um deles, usando a fórmula 2 (slides 17 e 18), para termos o tamanho do cromossomo.

**Gene 1**

$$2^k > (24 - 0) * 10^0$$

$$2^k > 24 * 1$$

$$2^k > 24$$

$$k = \log_2(24) = 4,585$$

Logo,  $k=5$  bits

**Gene 2**

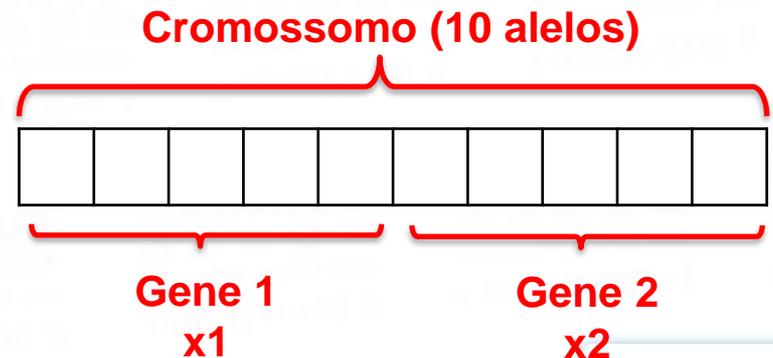
$$2^k > (16 - 0) * 10^0$$

$$2^k > 16 * 1$$

$$2^k > 16$$

$$k = \log_2(16) = 4$$

Mas,  $2^4 = 16 \dots$ ,  
*então*  $k=5$  bits



# Exercício 1 - Resolução

## Passo 2. Parametrização do AG

-Ponto(s) de corte: 1 após o 4º alelo –  
**Explicação: como os dois genes têm o mesmo tamanho. ponto de corte no meio do cromossomo só iria promover a troca de valores das variáveis x1 e x2**  
...

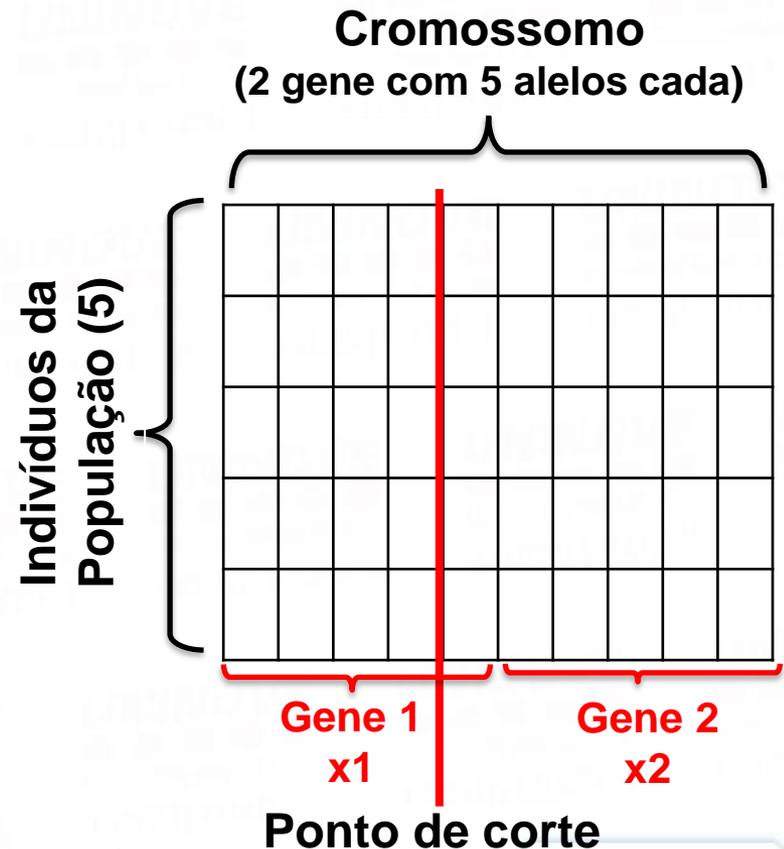
-Tamanho da população: 5 indivíduos

-Esquema de seleção: **Ranking – 3 melhores**

-Taxa de mutação: 10% (5 alelos serão mutados em cada geração)

## Passo 2. Execução do AG

Feita em planilha Excel (AG\_Probl\_Radios.xlsx)



## Exercício 2 – O problema dos 4 números

Como modelar um AG para encontrar os valores de A, B, C e D, de modo a satisfazer o sistema de equações abaixo?

$$\begin{cases} A + B + C + D = 7,11 \\ A * B * C * D = 7,11 \end{cases}$$

Esse problema pode ser formulado como um problema de otimização, como segue:

$$\textit{Minimizar } f(A, B, C, D) = |A + B + C + D - 7,11| + |A * B * C * D - 7,11|$$

Sujeito à:  $A, B, C, D \in \mathfrak{R}$

$$0,01 \leq A, B, C, D \leq 7,11$$

**Obs.: A simulação da execução do AG para resolver esse problema está na planilha Excel “AG\_Problema\_4Num.xls”.**

# Exercício 2 - Resolução

**Passo 1. OK** (não era naturalmente um probl. de otimização, mas foi formalizado corretamente um probl. de otimização)

## Passo 2. Modelagem do cromossomo

-Quantidade de genes do cromossomo: 4 genes (variáveis A, B, C e D: todas com  $\text{inf}=0,01$  e  $\text{sup}=7,11$ ; precisão de 2 casas decimais para todas as variáveis)

-Tamanho do cromossomo (com base no comprim. de cada gene): Embora sejam 4 genes, as variáveis que eles representam são iguais em termos de limites inferior/superior e precisão. Assim, calculando o comprimento de um gene, teremos o tamanho do cromossomo ( $\text{compr. gene} * 4$ )

### Gene 1 a 4

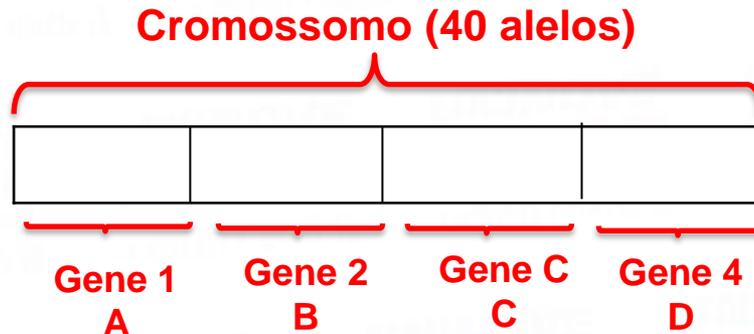
$$2^k > (7,11 - 0,01) * 10^2$$

$$2^k > 7,10 * 100$$

$$2^k > 710$$

$$k = \log_2(710) = 9,471$$

Logo,  $k=10$  bits



# Exercício 3 – Problema do Caminho mínimo (PCM)

Dado  $C = \{c_1, c_2, \dots, c_N\}$  o conjunto de  $N$  cidades, indicando a origem ( $s$ ) e o destino ( $g$ ), e uma matriz de adjacências  $d_{ij}$  onde  $d_{ij} = d(c_i, c_j)$ , uma maneira de se representar matematicamente o PCM é:

$$\text{Minimizar } z = \sum_{i=1}^N \sum_{j=1}^N d_{ij} x_{ij}$$

$$\text{Sujeito a: } \sum_{j=1}^N x_{sj} = 1$$

$$\sum_{i=1}^N x_{ij} - \sum_{k=1}^N x_{jk} = 0 \quad \forall j \in N - \{s, g\}$$

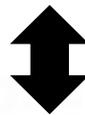
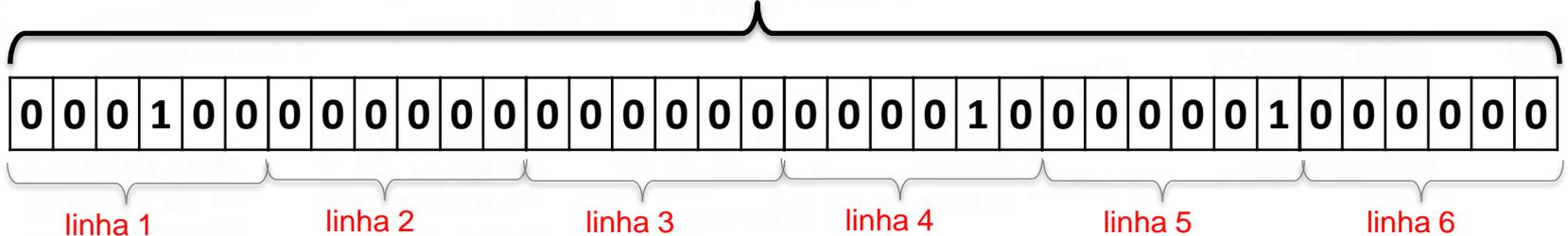
$$\sum_{i=1}^N x_{ig} = 1$$

$$x_{ij} \in \{0,1\}, \quad d_{ij} = \infty \text{ para todo } i = j$$

Onde:  $x_{ij} = \begin{cases} 1, & \text{se cidade } j \text{ pode ser alcançada a partir da cidade } i \\ 0, & \text{caso contrário} \end{cases}$

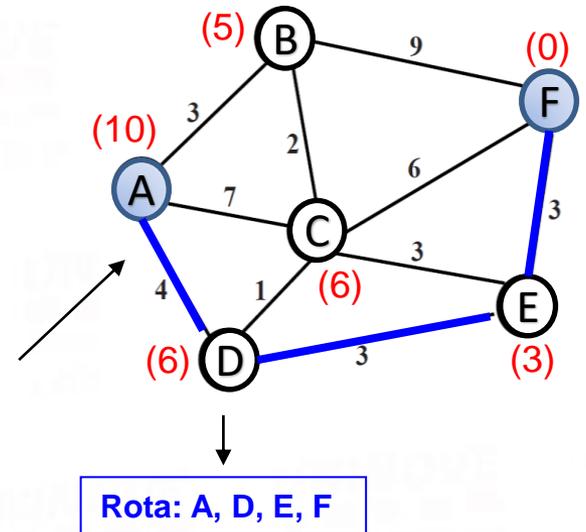
# Exemplo de aplicação do AG no PCM

## Cromossomo



Solução do Problema

	A	B	C	D	E	F
A	0	0	0	1	0	0
B	0	0	0	0	0	0
C	0	0	0	0	0	0
D	0	0	0	0	1	0
E	0	0	0	0	0	1
F	0	0	0	0	0	0



# Aplicações práticas

**Exploraremos aplicações práticas do AG nos seguintes contextos:**

- 1. Problema do Caminho Mínimo (solução alternativa às árvores de busca;**
- 2. Mineração de Dados (*datamining*) – descoberta de conhecimento em bases de dados**

**Haverá demonstração das aplicações práticas usando Python.**

# Referências

GOLDBARG, M. C. & LUNA, H. P., Otimização combinatória e programação linear: modelos e algoritmos, 3ª Edição. Rio de Janeiro: Editora Campus, 2000.

GOLDBERG, D. E. Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, 1989.

LINDEN, R. Algoritmos Genéticos. Brasport, 2006.

LUKE, S. Essentials of metaheuristics. 2. ed. Raleigh: Lulu, 2013.

MITCHELL, M. An Introduction to Genetic Algorithms. The MIT Press, 1998.

RUSSELL, S. J.; NORVIG, P. Inteligência artificial . 2. ed. Rio de Janeiro: Campus, 2004.

SOUZA, M. J. F. Inteligência Computacional para Otimização, Notas de aula, Departamento de Computação, Universidade Federal de Ouro Preto, disponível em: <http://www.decom.ufop.br/prof/marcone>.