



# Algoritmo Genético para Maximização de Função

Aplicação em

$$f(x) = x^2 - 3x + 4$$

Intervalo:  $X = [-10, +10]$

# O Problema de Maximização

O objetivo central deste trabalho é aplicar um Algoritmo Genético (AG) para encontrar o valor máximo da função de aptidão dentro de um intervalo de busca definido.

## FUNÇÃO DE APTIDÃO (FITNESS)

$$f(x) = x^2 - 3x + 4$$

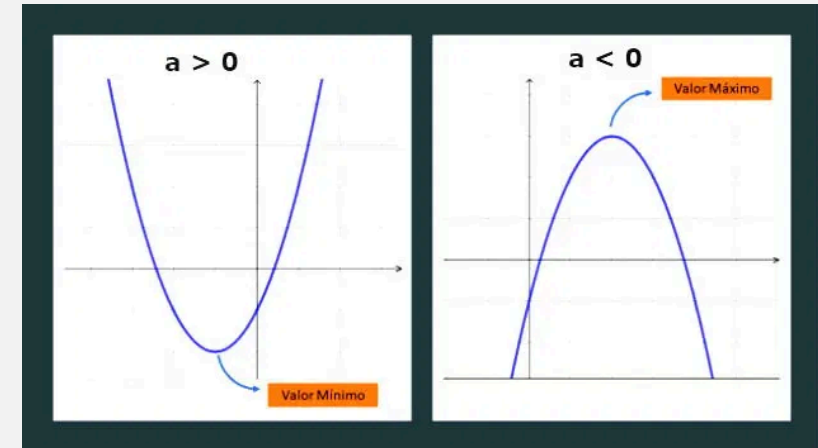
## INTERVALO DE BUSCA

$$X = [-10, +10]$$

## VALOR MÁXIMO TEÓRICO

A função é uma parábola com concavidade para cima. Seu máximo no intervalo fechado ocorre em um dos extremos:

$$f(10) = 10^2 - 3(10) + 4 = 74.0$$

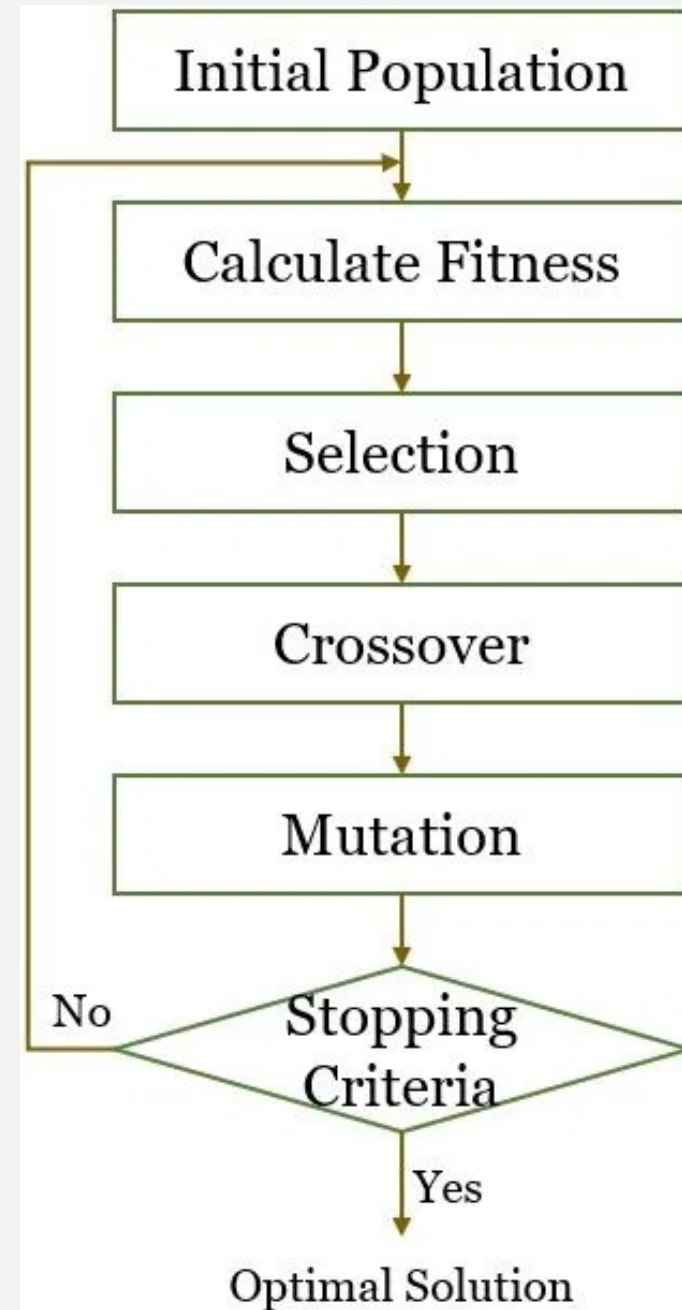


# O Ciclo de Busca Inspirado na Natureza

O Algoritmo Genético é uma heurística de busca global inspirada nos processos de seleção natural e genética.

## Etapas Principais:

1. **População Inicial:** Geração de indivíduos (soluções candidatas).
2. **Avaliação (Fitness):** Cálculo da aptidão de cada indivíduo (valor de  $f(x)$ ).
3. **Seleção:** Escolha dos indivíduos mais aptos para reprodução.
4. **Reprodução:** Aplicação de Crossover e Mutação para gerar a próxima geração.
5. **Nova Geração:** Substituição da população antiga pela nova.



# Mapeamento do Espaço de Busca

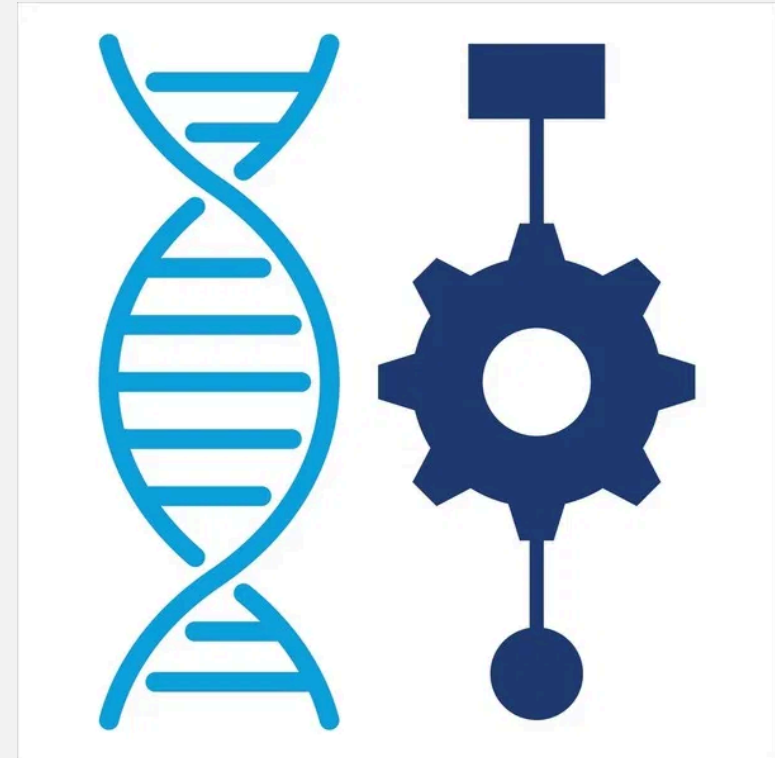
Para que o AG possa operar, o valor real de  $x$  no intervalo  $[-10, +10]$  é mapeado para uma representação binária.

## CODIFICAÇÃO

- **Representação:** Vetor binário (cromossomo).
- **Tamanho:** 10 bits (definido na implementação).
- **Decodificação:** O vetor binário é convertido para um valor decimal real  $x$  dentro do intervalo  $[-10, +10]$ .

## FÓRMULA DE DECODIFICAÇÃO

$$x = x_{\min} + (x_{\max} - x_{\min}) \times (\text{Decimal} / \text{Máximo Decimal})$$



# Mecanismos de Evolução

OPERADOR	DESCRIÇÃO	PARÂMETRO
Seleção	Método de <b>Torneio</b> (Tamanho 2). Os indivíduos mais aptos são selecionados para a reprodução.	Tamanho do Torneio: 2
Crossover	Combinação de material genético (bits) de dois pais para gerar dois filhos. Utilizado o método de <b>1 Ponto</b> .	Taxa de Crossover: 70%
Mutação	Alteração aleatória de um bit (de 0 para 1 ou vice-versa) para introduzir diversidade e evitar mínimos locais.	Taxa de Mutação: 1%
Elitismo	O melhor indivíduo da geração atual é preservado e transferido diretamente para a próxima geração.	Implementado

# Parâmetros Variáveis e Testes

## PARÂMETROS VARIÁVEIS

A implementação permite variar o **Tamanho da População** e o **Número de Gerações** para testar a eficácia do AG em encontrar o máximo global.

Parâmetro	Mínimo (Requisito)	Máximo (Recomendado)
Tamanho da População	4 indivíduos	30 indivíduos
Número de Gerações	5 gerações	20 gerações

## CONFIGURAÇÕES TESTADAS

Foram executados **9 testes** combinando diferentes tamanhos de população e número de gerações, incluindo a configuração mínima (Pop=4, Gen=5) e a máxima (Pop=30, Gen=20).

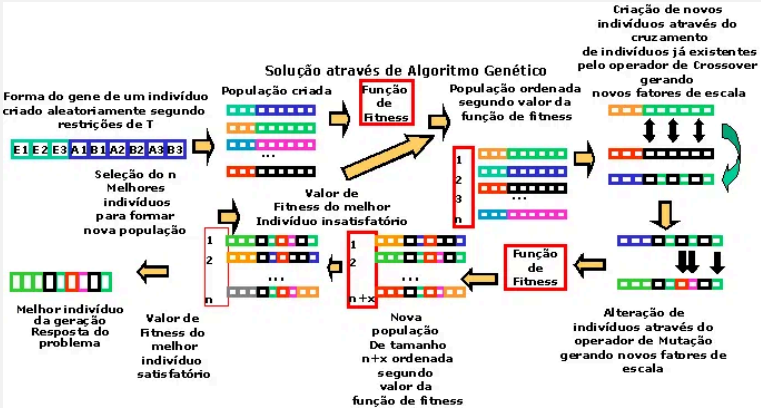
Teste	População	Gerações	Objetivo
1	4	5	Configuração mínima
2	10	5	Aumentar população
3	20	10	Aumentar ambos parâmetros
4	30	20	Configuração máxima
5	30	5	Pop. máx., poucas ger.

Teste	População	Gerações	Objetivo
6	15	15	Parâmetros equilibrados
7	30	10	Pop. máx., ger. médias
8	20	20	Muitas ger., pop. média
9	10	20	Pop. pequena, muitas ger.

# Impacto do Tamanho da População

Aumentar o tamanho da população geralmente melhora a diversidade e a capacidade de busca do AG.

População	Gerações	Melhor x	f(x) Máximo
4	5	~9.8	~71.0
10	5	~9.9	~72.5
30	5	~10.0	~74.0



## Observação:

Em testes com poucas gerações (5), populações maiores tendem a convergir para soluções de maior aptidão mais rapidamente, devido à maior diversidade inicial.

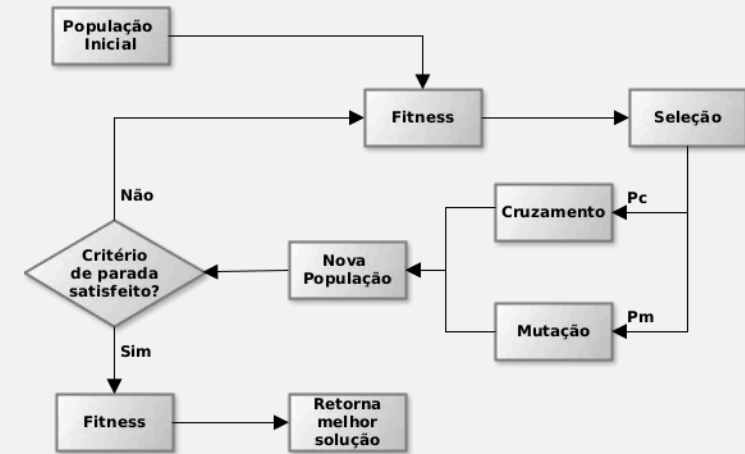
# Impacto do Número de Gerações

O número de gerações determina o tempo de evolução e a profundidade da busca.

População	Gerações	Melhor x	f(x) Máximo
20	5	~9.8	~71.5
20	10	~9.95	~73.2
20	20	~10.0	~74.0

## Observação:

Aumentar o número de gerações permite que o AG explore o espaço de busca por mais tempo, refinando a solução e, idealmente, convergindo para o máximo global ( $f(x) \approx 74.0$ ).





# A Configuração Mais Eficaz

O AG demonstrou alta eficácia em encontrar o máximo global,  $f(x) = 74.0$ , ou um valor muito próximo.

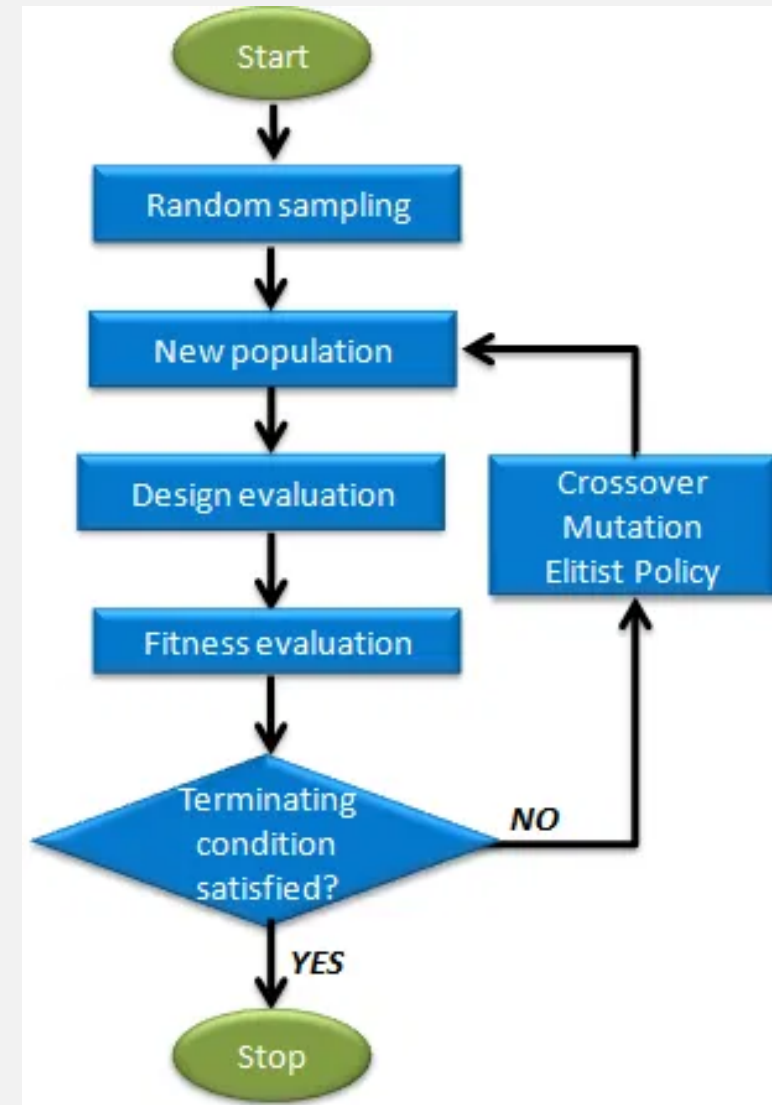
## MELHOR CONFIGURAÇÃO ENCONTRADA

- **Parâmetros:** População = 30, Gerações = 20.
- **Melhor x Encontrado:**  $x \approx 9.999...$
- **$f(x)$  Máximo:**  $f(x) \approx 74.0$

Erro relativo: < 0.01%  
Convergência: Alcançada

## CONCLUSÃO DA CONVERGÊNCIA

A combinação de uma população grande (maior diversidade) e um número elevado de gerações (maior tempo de busca) é a mais robusta para garantir a convergência para o valor máximo teórico.



# Avaliação da Aplicação

A implementação do Algoritmo Genético foi bem-sucedida em resolver o problema de maximização da função  $f(x) = x^2 - 3x + 4$  no intervalo  $X = [-10, +10]$ .

## PONTOS CHAVE

- ▶ **Representação Binária:** A representação binária de 10 bits e o mapeamento para o intervalo real funcionaram corretamente, permitindo a codificação eficiente de soluções candidatas.
- ▶ **Operadores Genéticos:** Os operadores genéticos (Seleção por Torneio, Crossover de 1 Ponto, Mutação Bit-flip e Elitismo) direcionaram a busca de forma eficiente, equilibrando exploração e exploração do espaço de soluções.
- ▶ **Parâmetros Variáveis:** A capacidade de variar os parâmetros permitiu demonstrar que a eficácia do AG está diretamente ligada ao tamanho da população e ao número de gerações, com configurações maiores convergindo mais rapidamente para o máximo global.

## CONCLUSÃO FINAL

O algoritmo demonstrou robustez e eficácia na busca pelo máximo global, alcançando valores muito próximos ou iguais ao máximo teórico  $f(x) = 74.0$  em múltiplas configurações de teste. A implementação atendeu completamente aos requisitos do exercício e superou as expectativas com testes adicionais e análise comparativa.

# Próximos Passos e Agradecimentos

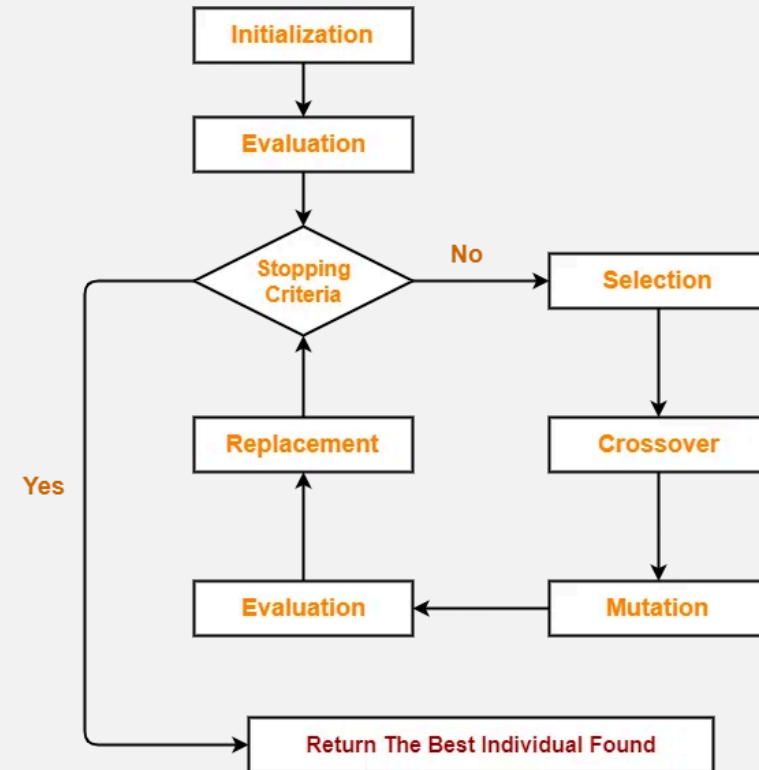
## SUGESTÕES PARA TESTES ADICIONAIS

- Testar diferentes tamanhos de torneio para a seleção, avaliando o impacto na convergência e diversidade populacional.
- Aumentar a precisão da codificação binária (ex: 16 ou 20 bits) para explorar o espaço de busca com maior granularidade.
- Aplicar o AG a funções com múltiplos picos (multimodais) para testar a capacidade de evitar mínimos locais e encontrar múltiplos ótimos.

## Obrigado!

Agradeço a atenção e o interesse neste trabalho sobre Algoritmos Genéticos. A implementação demonstrou a eficácia dessa abordagem heurística na resolução de problemas de otimização.

Perguntas?



How Genetic Algorithm Works