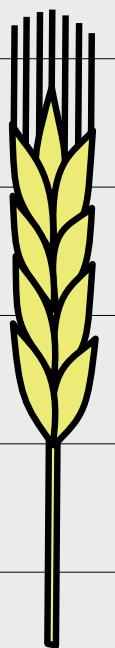
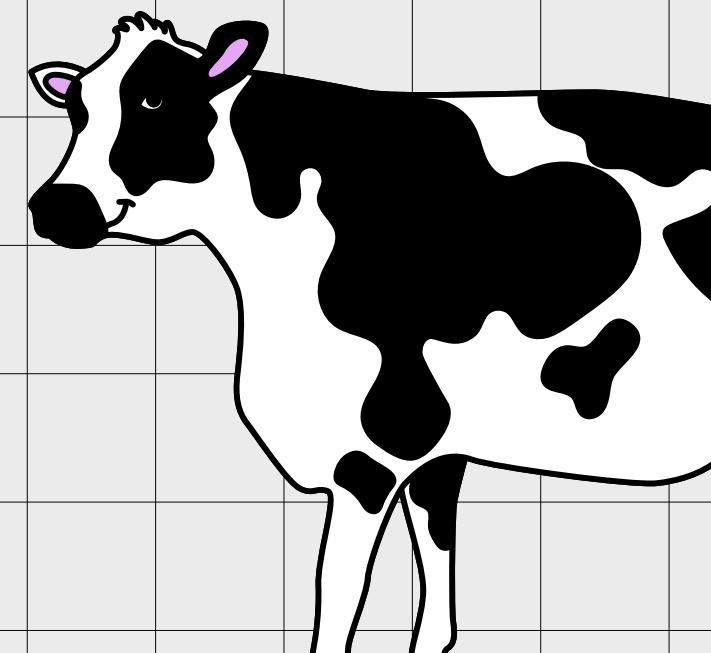
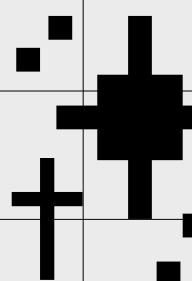
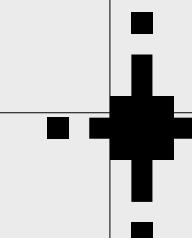
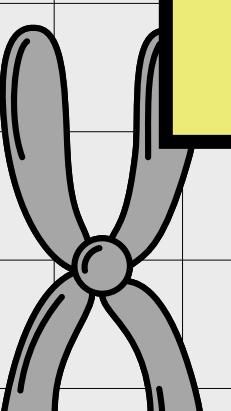
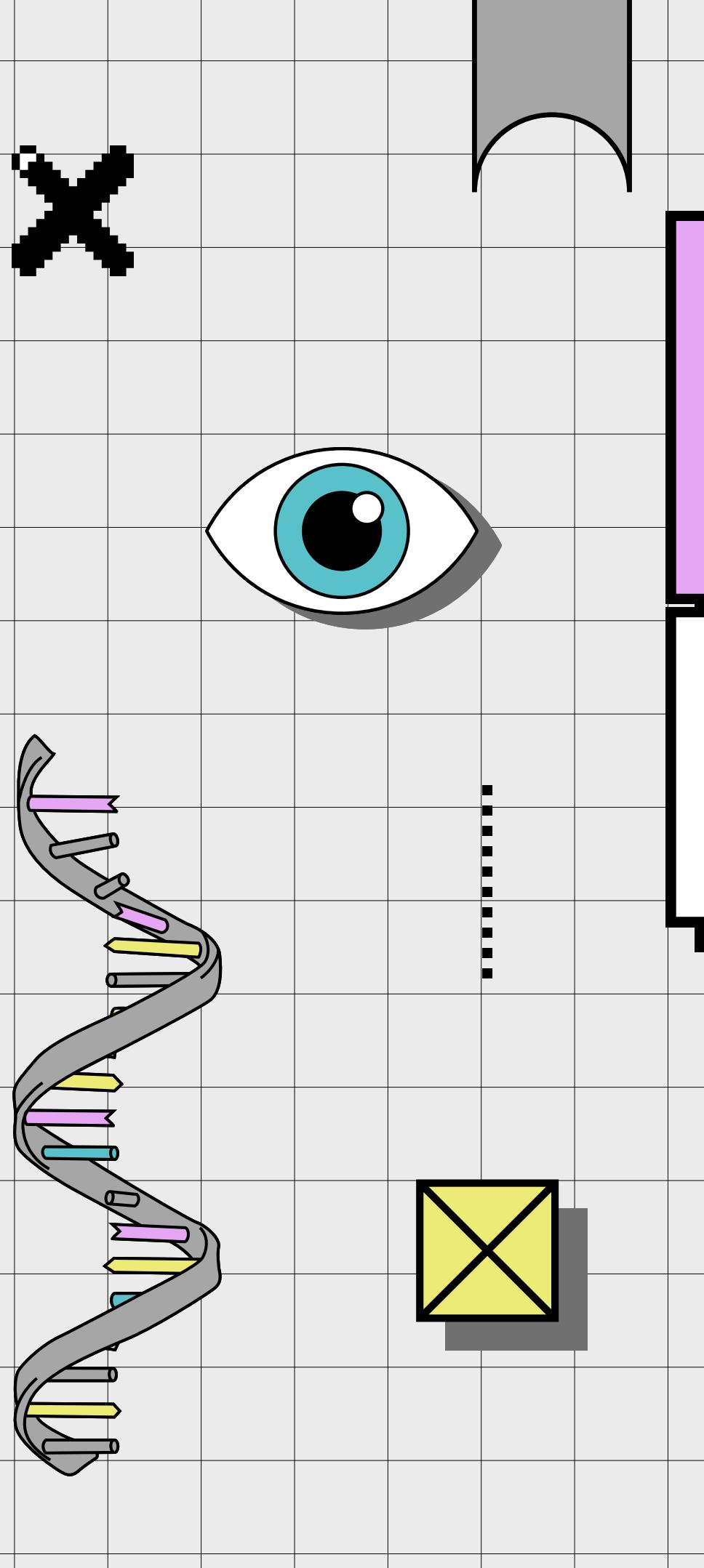
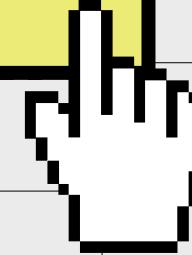


ALGORITMO GENÉTICO

Aplicado ao problema da
mochila binária

Iury Alcantara
Marcus Mariano
Mateus Ramos
Vitor Baza



A META-HEURÍSTICA

- Criada em 1960, por Jhon Holland;
- Baseada no conceito de evolução de Charles Darwin;
- Ótimas para grandes espaços de solução;
- Possuem variabilidade, evitando vícios e atendendo todo tipo de abordagem;



COMPONENTES

Partes do algoritmo utilizadas para simular o processo evolutivo

SAMPLE ANSWER

CROMOSSOMOS

São os conjuntos de soluções possíveis para o problema.

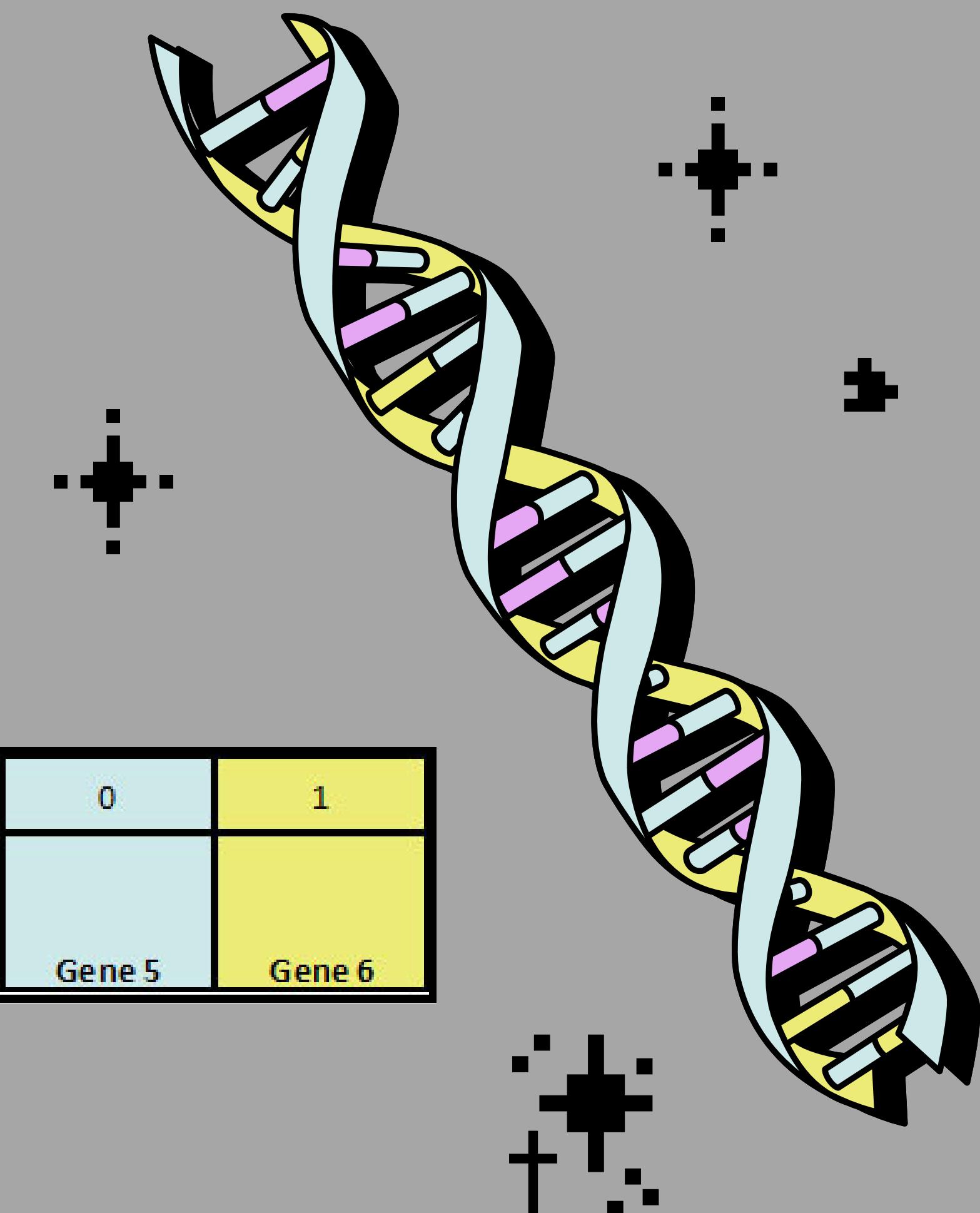
Geneticamente, são os indivíduos que se reproduzirão para formar novos indivíduos, novas soluções.

Características	A	B	C	D	E	F
Cromossomo 1 ?	0	1	0	1	0	1
Cromossomo 2 ?	1	0	0	1	0	1
Cromossomo 4 ?	1	1	1	0	1	0
Cromossomo 5 ?	1	0	0	0	0	0
Cromossomo 6 ?	0	0	1	0	1	0

GENES

São as representações de uma característica ou parâmetro da solução específica que o algoritmo está tentando otimizar.

Cromossomo 1 ?	0	1	0	1	0	1
	Gene 1	Gene 2	Gene 3	Gene 4	Gene 5	Gene 6

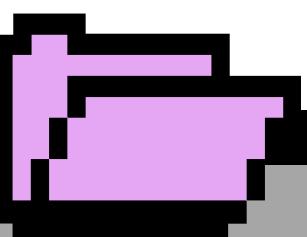


POPULAÇÃO

SAMPLE ANSWER

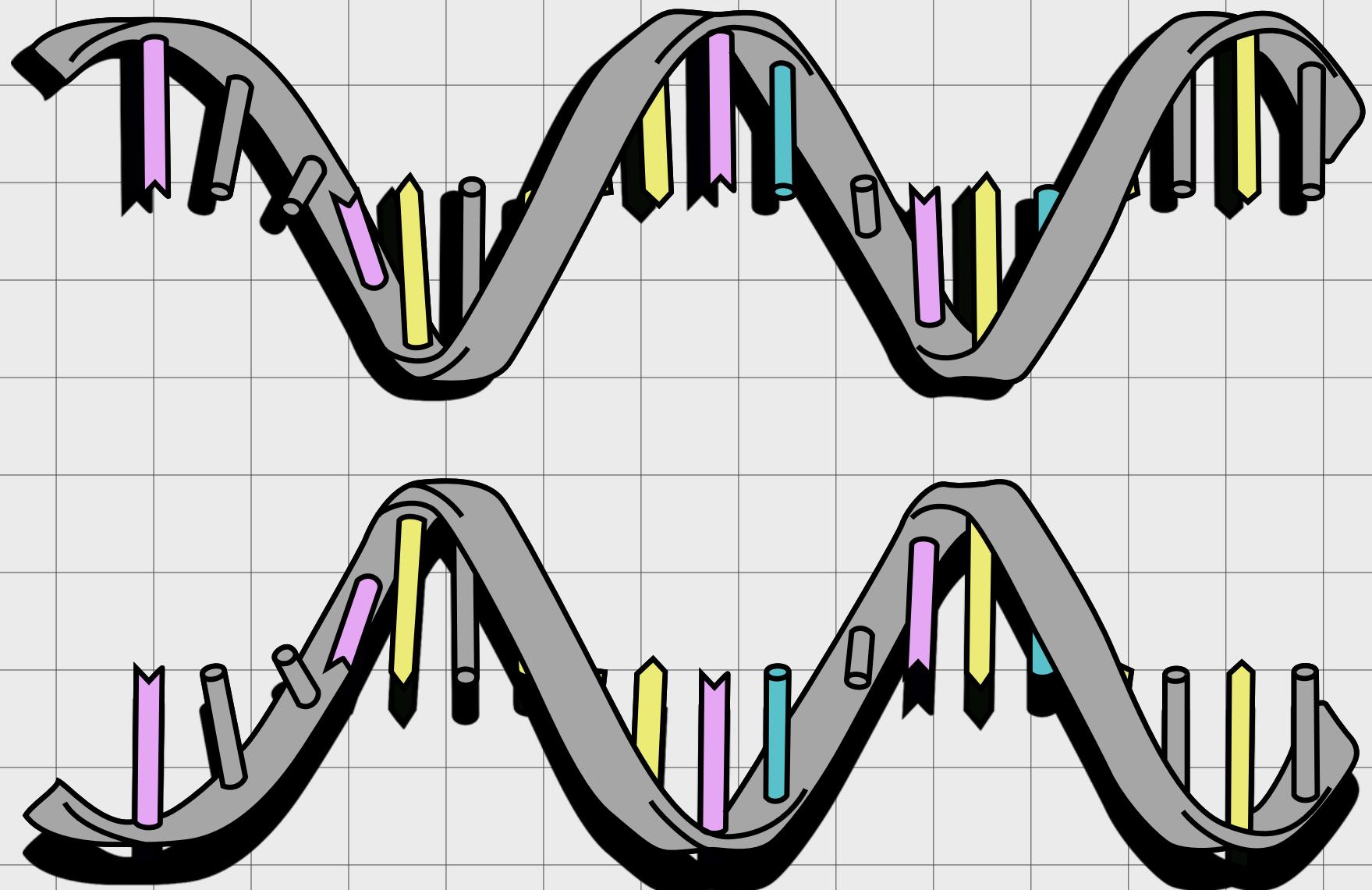
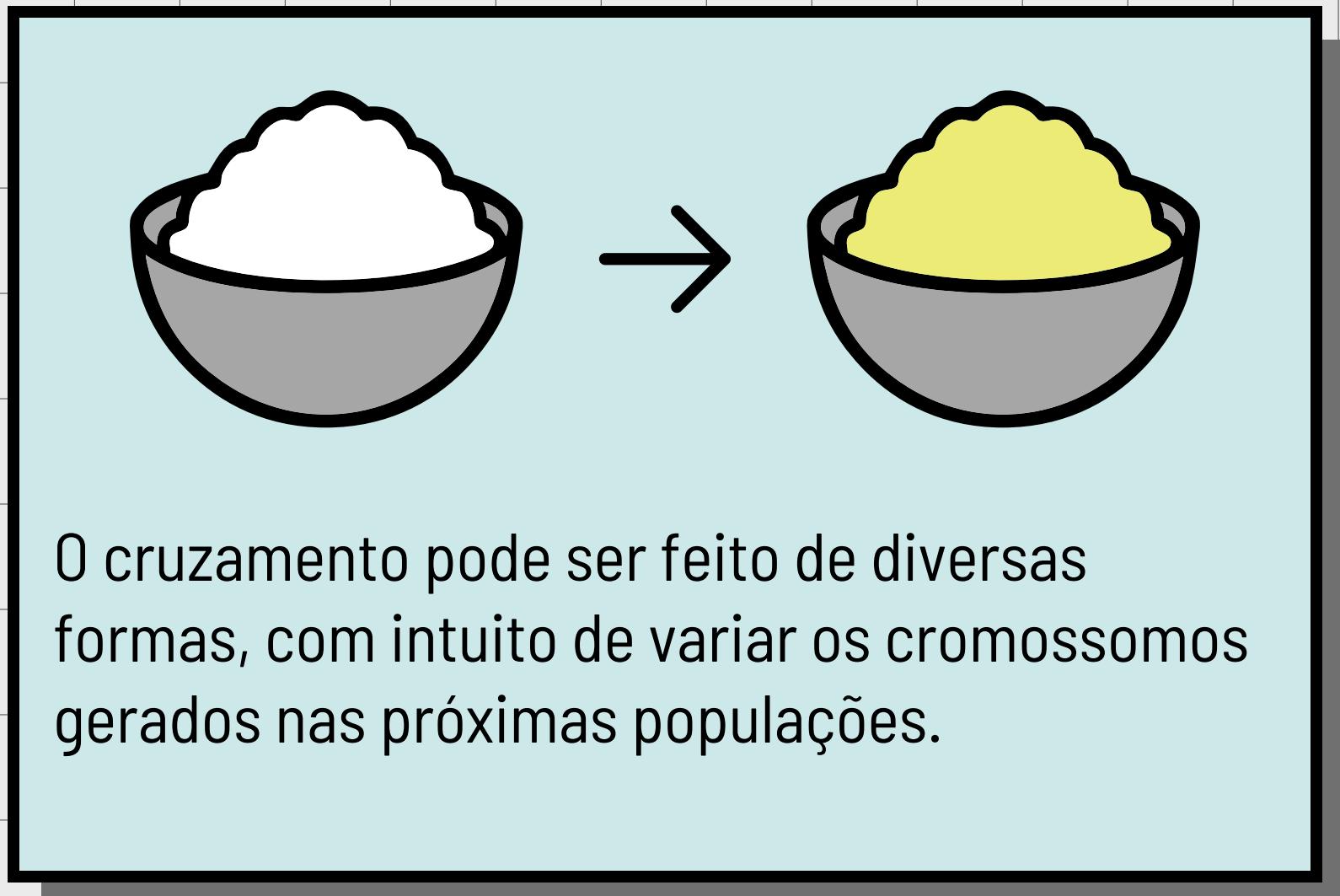
A população é o conjunto de soluções, ou cromossomos, que existem em uma geração específica.

A população pode explorar o espaço de soluções com essa dinâmica e, idealmente, chegar a uma solução otimizada ao longo do tempo.



CRUZAMENTO

São responsáveis pela reprodução sexuada, que consiste na combinação de genes de cromossomos para produzir um ou mais descendentes



CRUZAMENTO POR CORTE

Características	A	B	C	D	E	F
Cromossomo 1 ?	0	1	0	1	1	1
Ponto de Corte		↑			↓	
Cromossomo 2 ?	0	0	0	1	0	1
Cromossomo Gerado ?	0	1	0	1	0	1

CRUZAMENTO POR SORTEIO

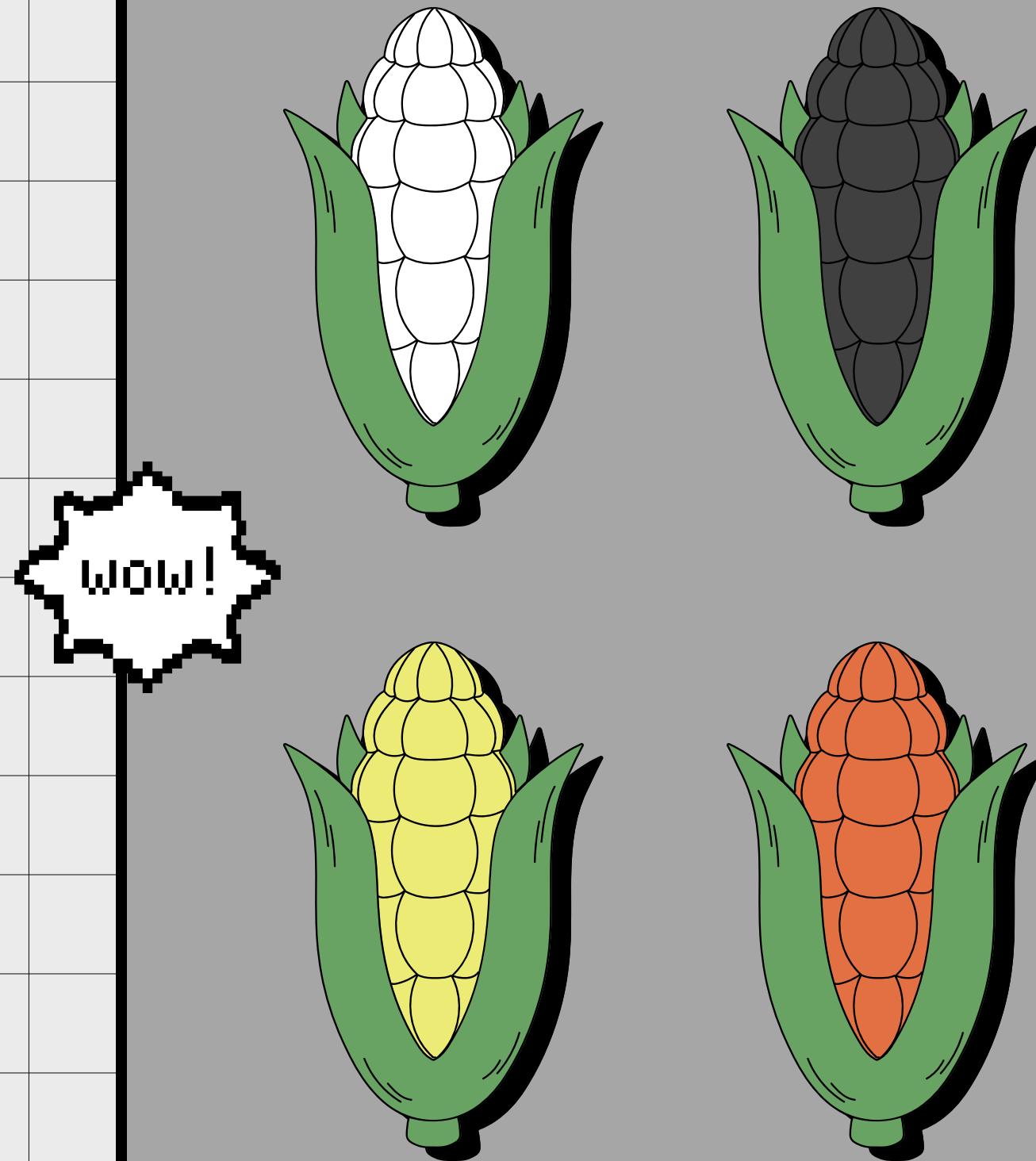
Características	A	B	C	D	E	F
Cromossomo 1 ?	0	1	0	1	1	1
Sorteio	0,19	0,61	0,38	0,28	0,88	0,98
Cromossomo 2 ?	0	0	0	0	0	1
Cromossomo Gerado ?	0	0	0	1	0	1

MUTAÇÃO

A mutação é outra forma de variar e melhorar os resultados, evitando que fiquem presos em resultados sub-ótimos.

São escolhidos alguns genes aleatórios de cada cromossomo para terem seus valores trocados.

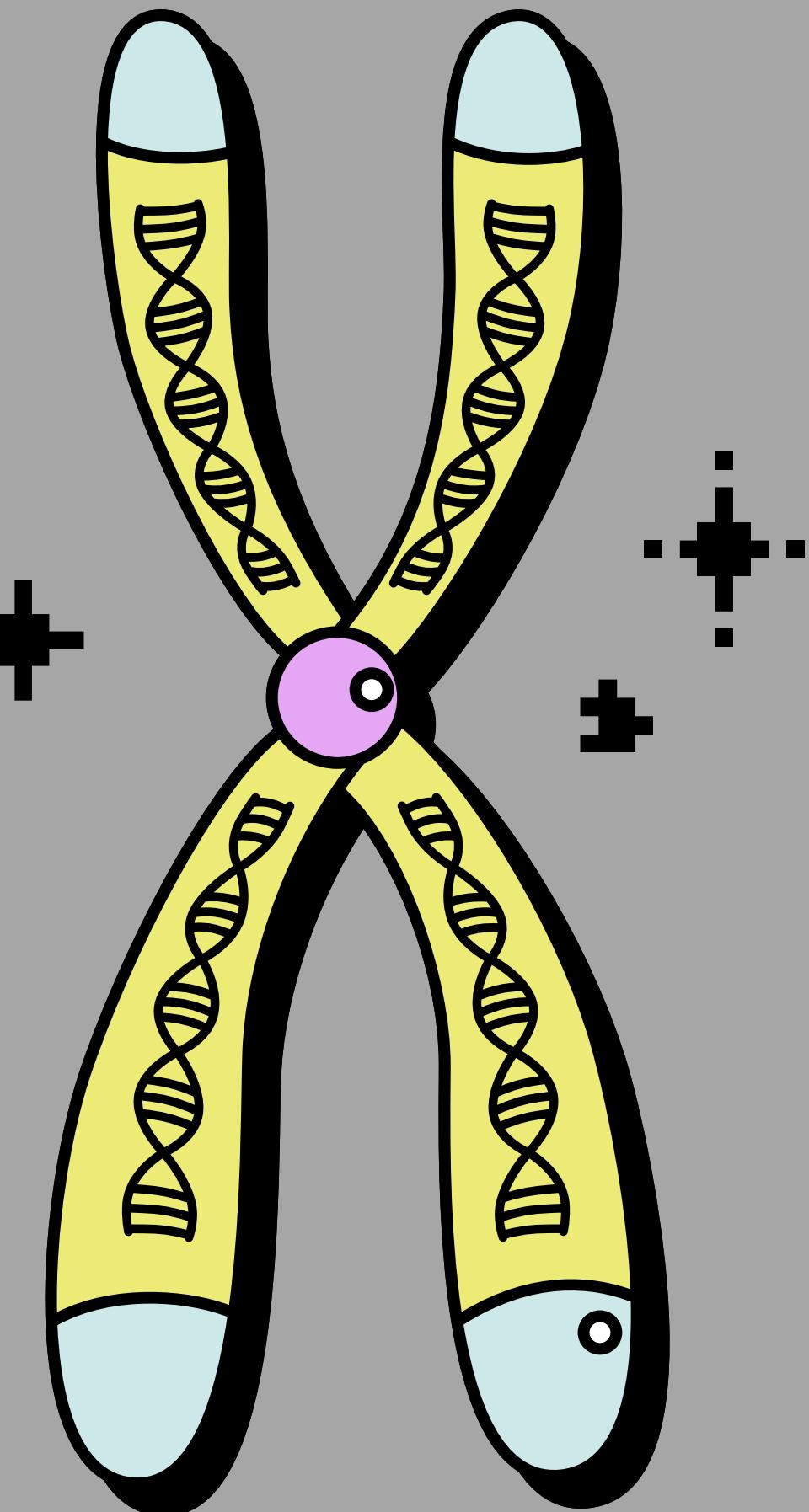
Garante a diversidade genética da população.



FUNÇÃO DE APTIDÃO

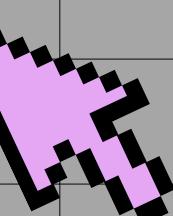
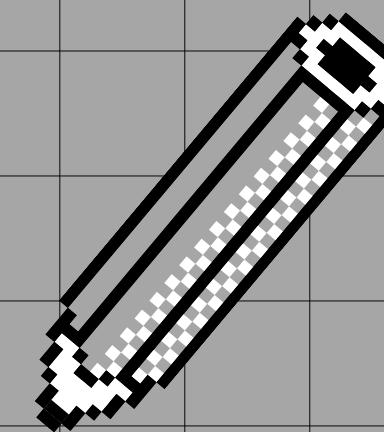
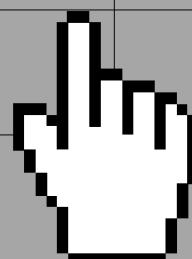
Avalia se o cromossomo cumpre as regras para fazer parte daquela população.

Qualifica e ordena os cromossomos conforme o seu valor de aptidão.



CRITÉRIO DE PARADA

- 1 Número Máximo de Gerações
- 2 Convergência da População
- 3 Qualidade de Solução



Algoritmo Genético Mochila

// Inicialização

População <- GerarPopulaçãoInicialAleatória()

AvaliarPopulação(População)

DividirPopulação(População, Elite, Normais, Ralé)

// Loop principal

Enquanto (condição de parada não atingida) **fazer**

NovaPopulação <- []

// Reprodução Elite x Elite

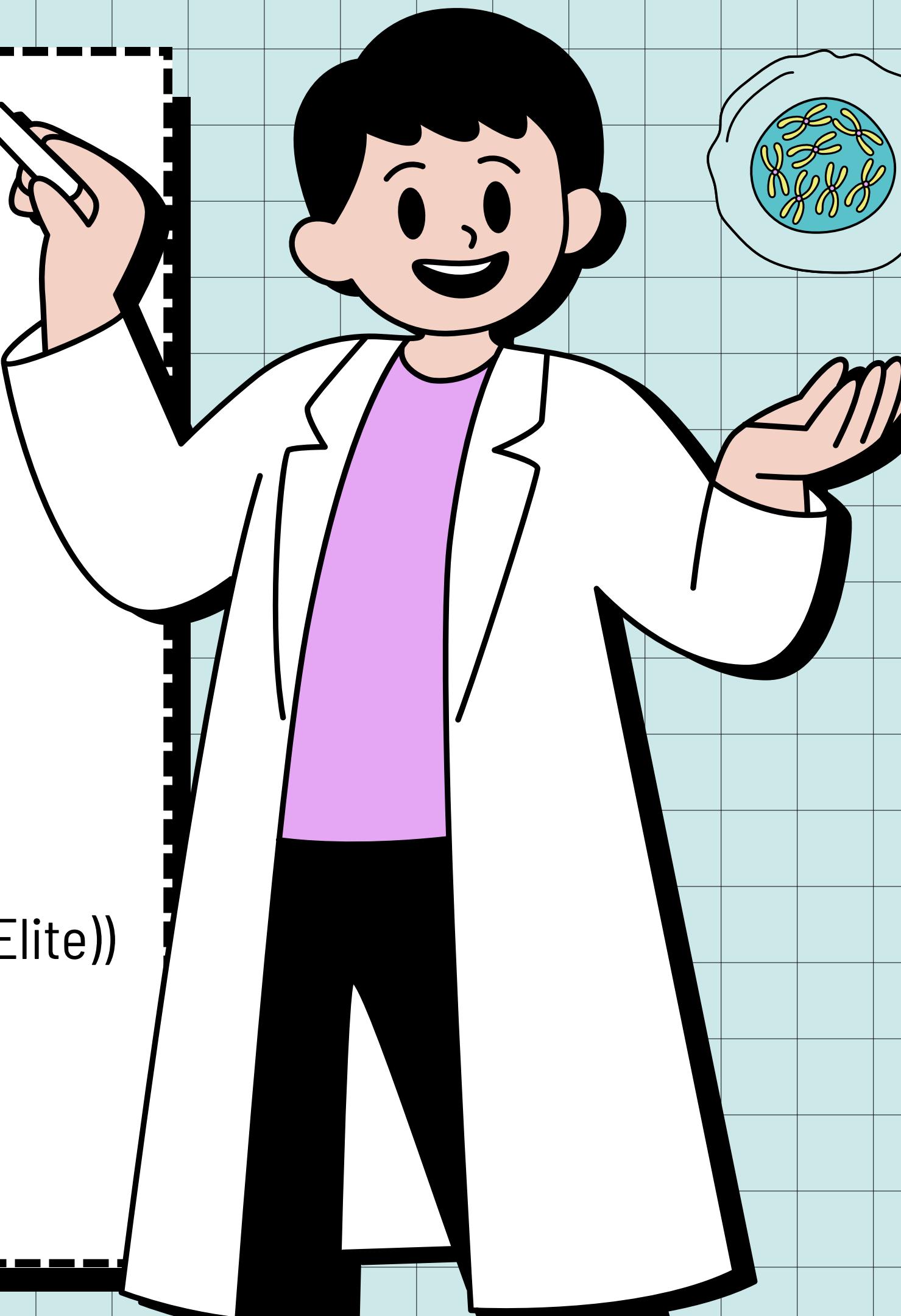
Para cada indivíduo da Elite[] **fazer**

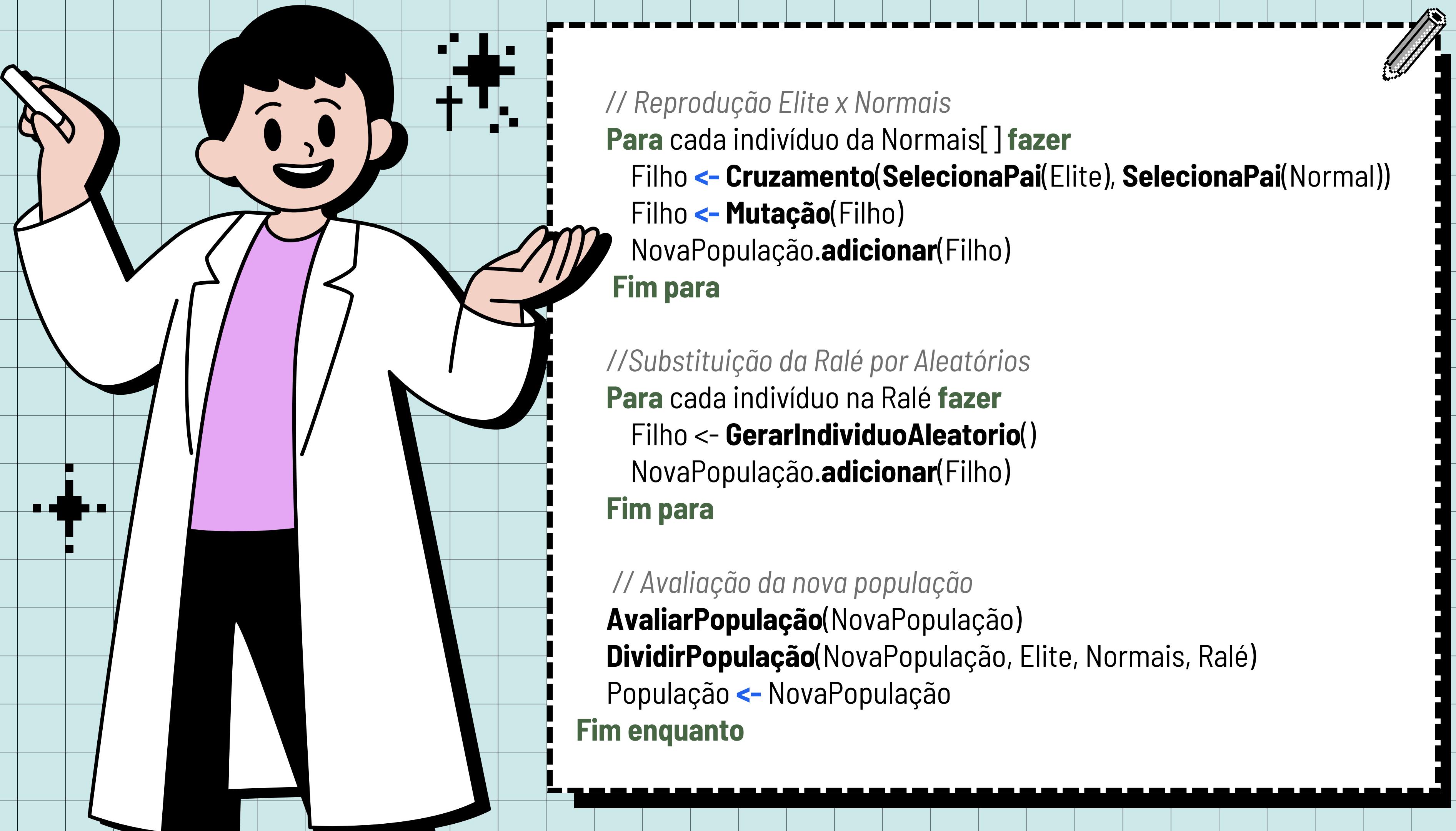
Filho <- Cruzamento(SelecionaPai(Elite), SelecionaPai(Elite))

Filho <- Mutação(Filho)

NovaPopulação.adicionar(Filho)

Fim para





// Reprodução Elite x Normais

Para cada indivíduo da Normais[] **fazer**

 Filho <- **Cruzamento**(**SelecionaPai**(Elite), **SelecionaPai**(Normal))

 Filho <- **Mutação**(Filho)

 NovaPopulação.**adicionar**(Filho)

Fim para

// Substituição da Ralé por Aleatórios

Para cada indivíduo na Ralé **fazer**

 Filho <- **GerarIndividuoAleatorio**()

 NovaPopulação.**adicionar**(Filho)

Fim para

// Avaliação da nova população

AvaliarPopulação(NovaPopulação)

DividirPopulação(NovaPopulação, Elite, Normais, Ralé)

População <- NovaPopulação

Fim enquanto

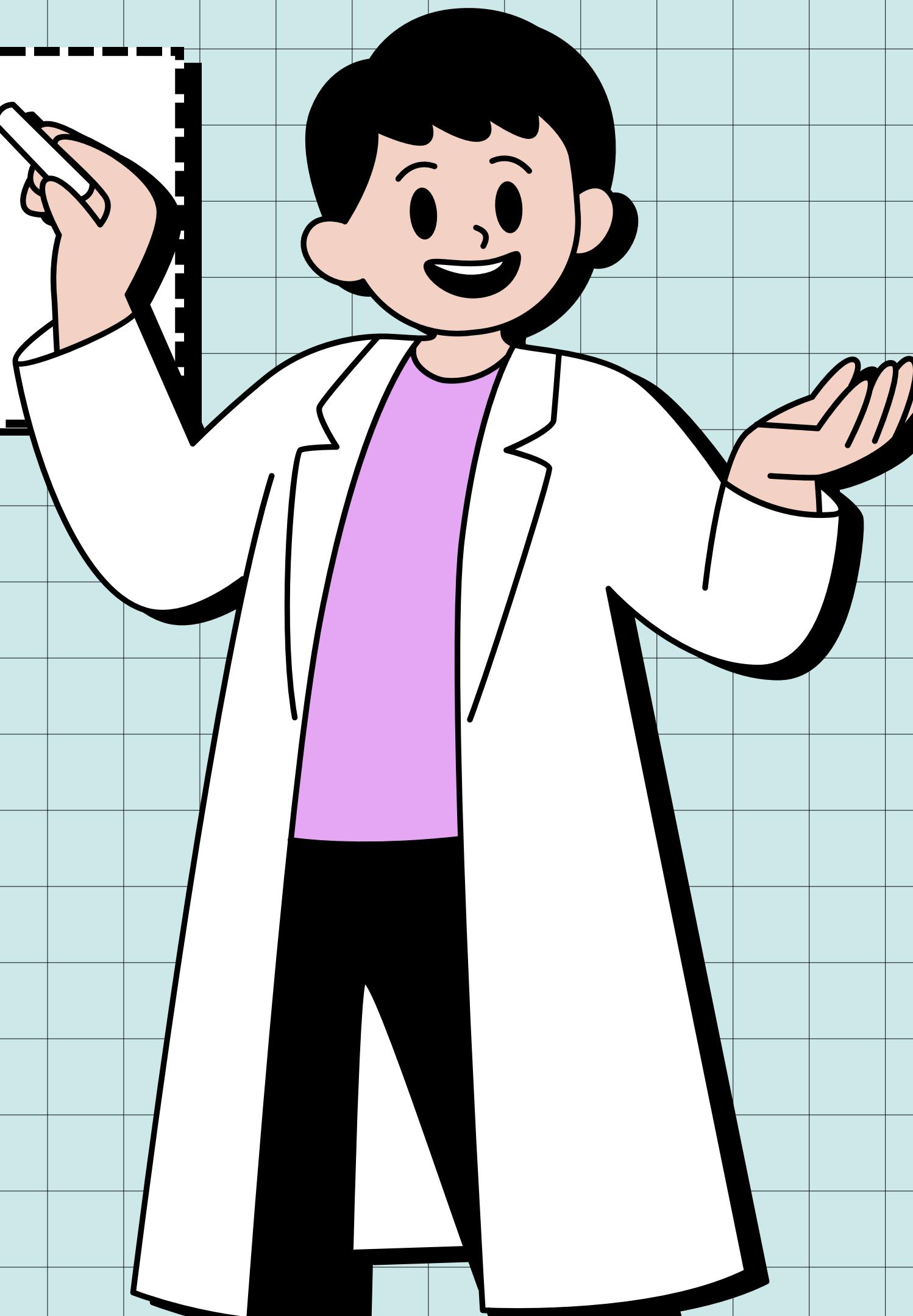
// Retornar a melhor solução encontrada

Retornar MelhorIndivíduo(População)

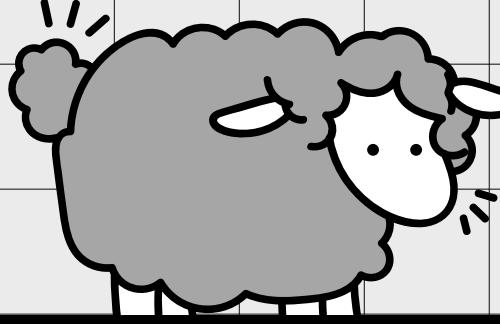
Fim Algoritmo



wow!



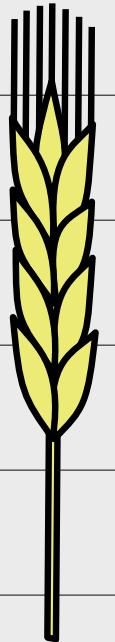
ESPAÇO AMOSTRAL



Item	1	2	3	4	5	6	7
Peso (kg)	6	3	1	7	4	2	5
Valor (R\$)	2	7	3	4	5	2	6

População 01	Indivíduo 01	0	1	0	1	0	1	0
	Indivíduo 02	1	0	0	0	1	0	1
	Indivíduo 03	1	0	1	0	0	1	0

Peso (kg)	Valor (R\$)
12	13
15	13
9	7



ALGORITMO GENÉTICO X FORÇA BRUTA

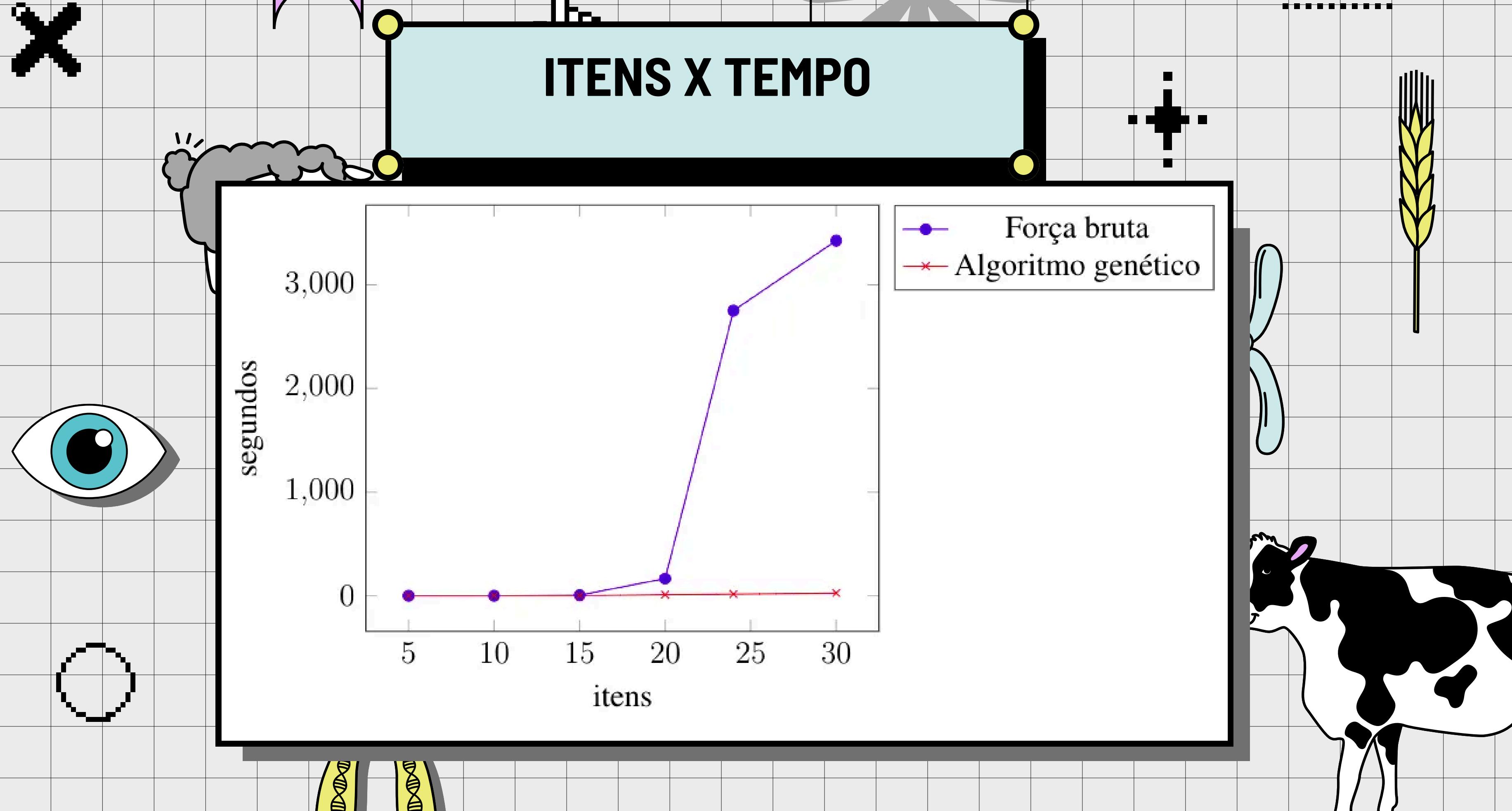
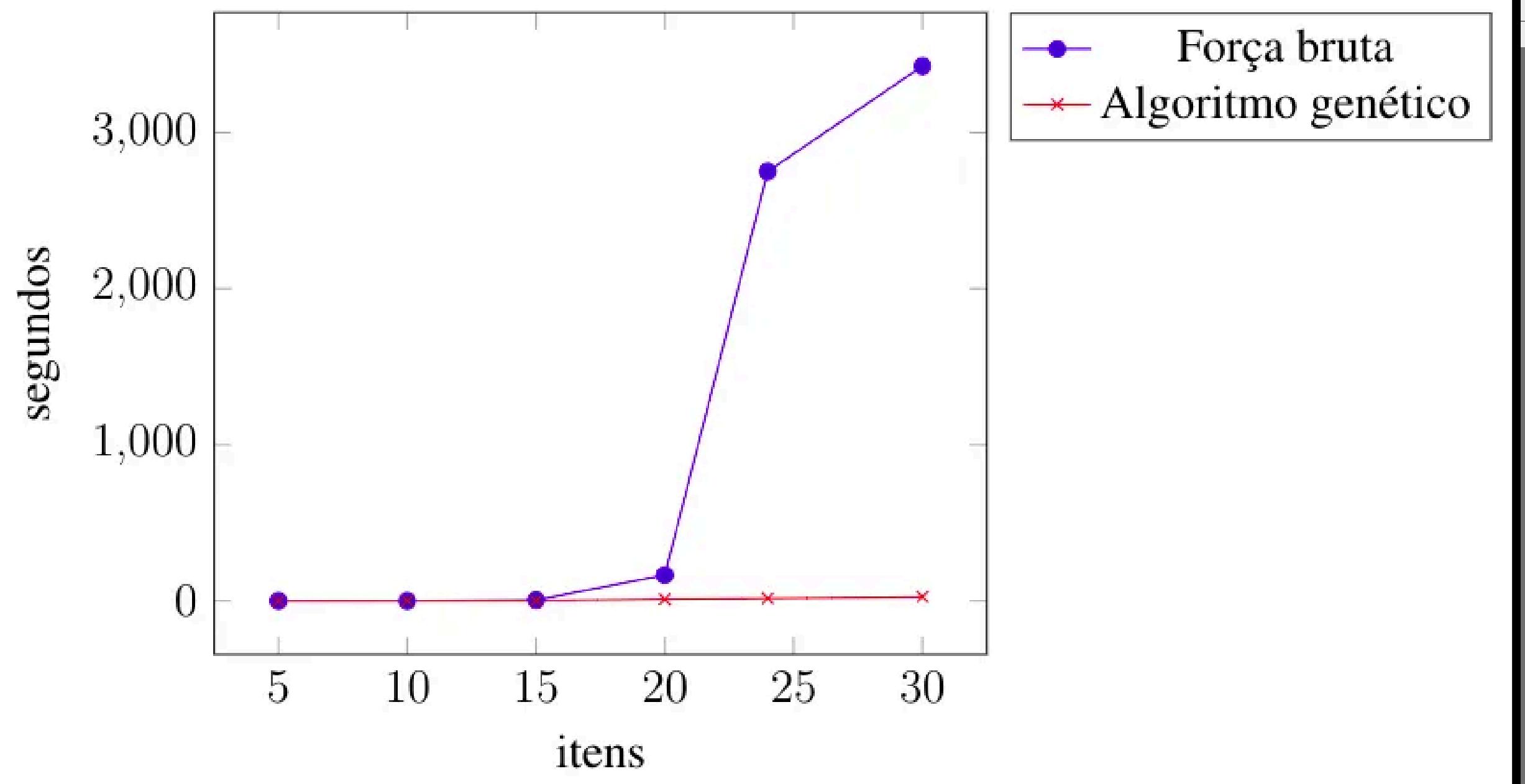
Search this file...

	Itens	Força Bruta (seg.)	Algoritmo Genético (seg.)	Gerações
1	5	0.005731	0.013566	5
2	10	0.229117	0.335321	161
3	15	5.376002	1.458588	611
4	20	165.451445	10.469887	3733

knapsack-results.csv hosted with ❤ by GitHub

[view raw](#)

ITENS X TEMPO



QUALIDADE DA SOLUÇÃO

Search this file...

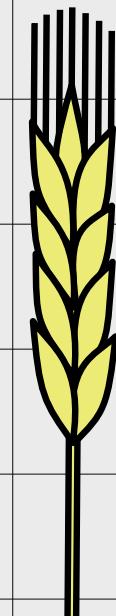
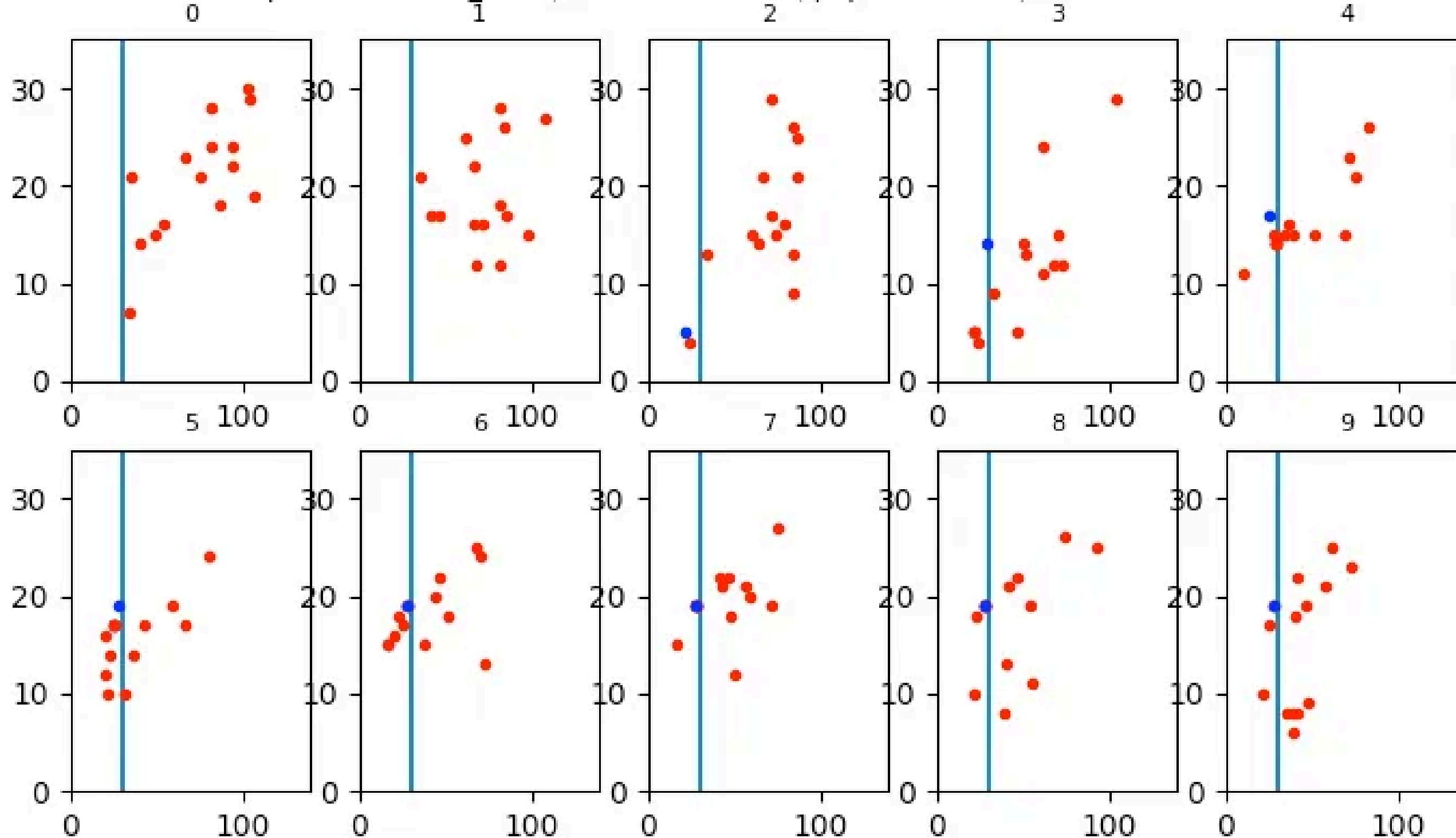
Itens	Força Bruta	Algoritmo Genético	Aproximação
5	306	306	100%
10	848	844.8	99.62%
15	1458	1444.8	99.09%
20	12585215	11688599	92.87%

knapsack-results2.csv hosted with ❤ by GitHub

[view raw](#)

MELHOR INDIVÍDUO

Knapsack Problem
procreate: first_elite | mutate: random | population: 10 | individuals: 15



OBRIGADO



SANTOS, Philippe Leal Freire dos. "**Heurísticas para o Problema da Partição Cromática de Custo Mínimo**". 2018. Disponível em: <https://site.ic.uff.br/wp-content/uploads/2021/09/872.pdf>. Acesso em: 16 fev. 2023.

Algoritmos Genéticos. Disponível em: <<https://sites.icmc.usp.br/andre/research/genetic/>>. Acesso em: 25 ago. 2024.

Algoritmos Genéticos Aplicados Ao Problema Da Mochila Binária. Disponível em: <<https://pt.scribd.com/document/715911860/Algoritmos-Geneticos-Aplicados-Ao-Problema-Da-Mochila-Binaria>>. Acesso em: 25 ago. 2024.

BATISTA, V. E. **Algoritmo Genético para o Problema da Mochila** - Vitor Emanuel Batista. Disponível em: <<https://vitorebatista.medium.com/algoritmo-gen%C3%A9tico-para-o-problema-da-mochila-5910f90f9488>>. Acesso em: 25 ago. 2024.

CARVALHO, Rubens. **Problema da Mochila**. Disponível em: <<https://www.ime.unicamp.br/~mac/db/2015-1S-122181-1.pdf>>. Acesso em: 25 ago. 2024.

PACHECO, Marco Aurélio Cavalcanti. **Algoritmos genéticos: princípios e aplicações**. Disponível em: <https://inf.ufsc.br/~mauro.roisenberg/ine5377/Cursos-ICA/CE-intro_apost.pdf>. Acesso em: 25 ago. 2024.