

INTELIGÊNCIA ARTIFICIAL

BUSCA LOCAL E
OTIMIZAÇÃO

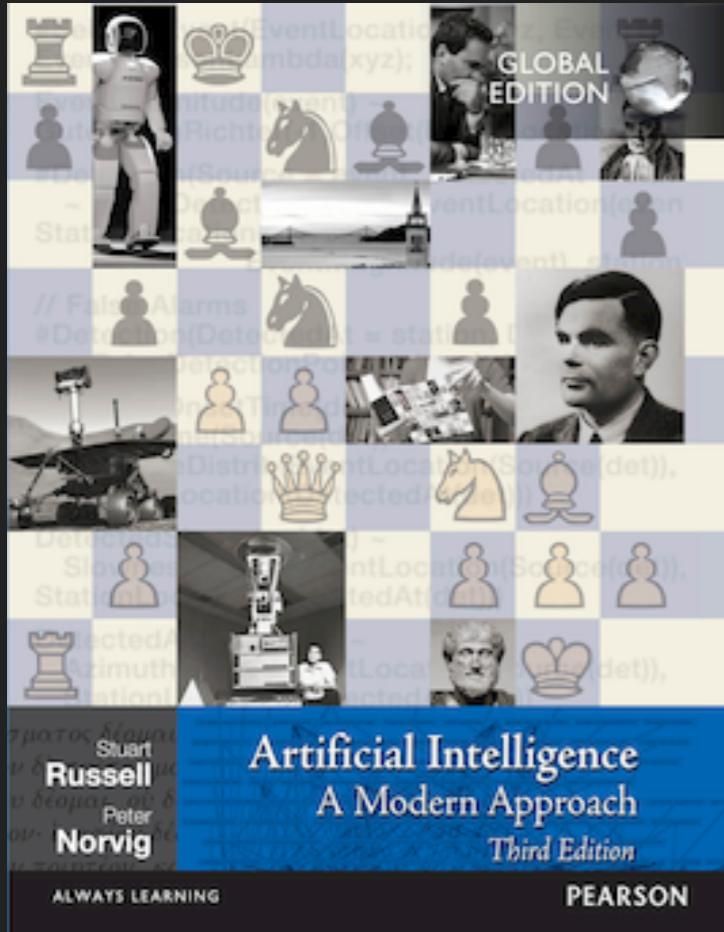
Lucas Baggio Figueira [@lucasfigueira]

MATERIAL DE ESTUDO

Artificial Intelligence: A modern approach

Cap. 4

MATERIAL DE ESTUDO



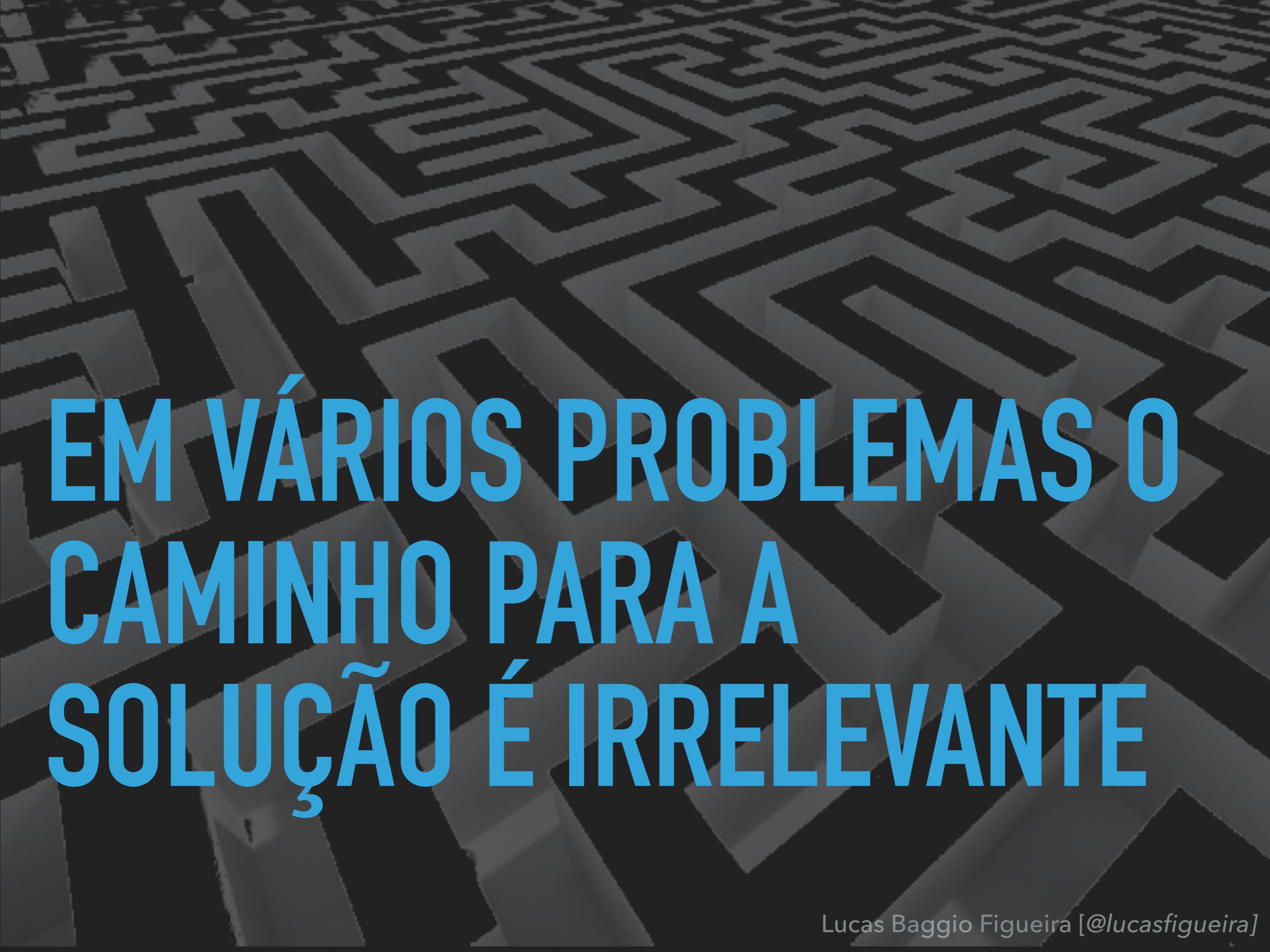
Artificial Intelligence: A modern approach

Cap. 4

SUMÁRIO

- ▶ Subida da Montanha (Hill-Climbing)
- ▶ Recozimento simulado (Simulated Annealing)
- ▶ Algoritmos Genéticos*

*brevemente



EM VÁRIOS PROBLEMAS
CAMINHO PARA A
SOLUÇÃO É IRRELEVANTE



O OBJETIVO QUE IMPORTA



ALGORITMOS DE MELHORIA ITERATIVA

ALGORITMOS DE MELHORIA ITERATIVA

- ▶ **Caminho irrelevante, apenas o estado final importa.**

ALGORITMOS DE MELHORIA ITERATIVA

- ▶ **Caminho irrelevante, apenas o estado final importa.**
- ▶ O espaço de busca pode ser num,

ALGORITMOS DE MELHORIA ITERATIVA

- ▶ **Caminho irrelevante, apenas o estado final importa.**
- ▶ O espaço de busca pode ser num,
 - ▶ conjunto de configurações, onde a melhor satisfaz certas restrições.

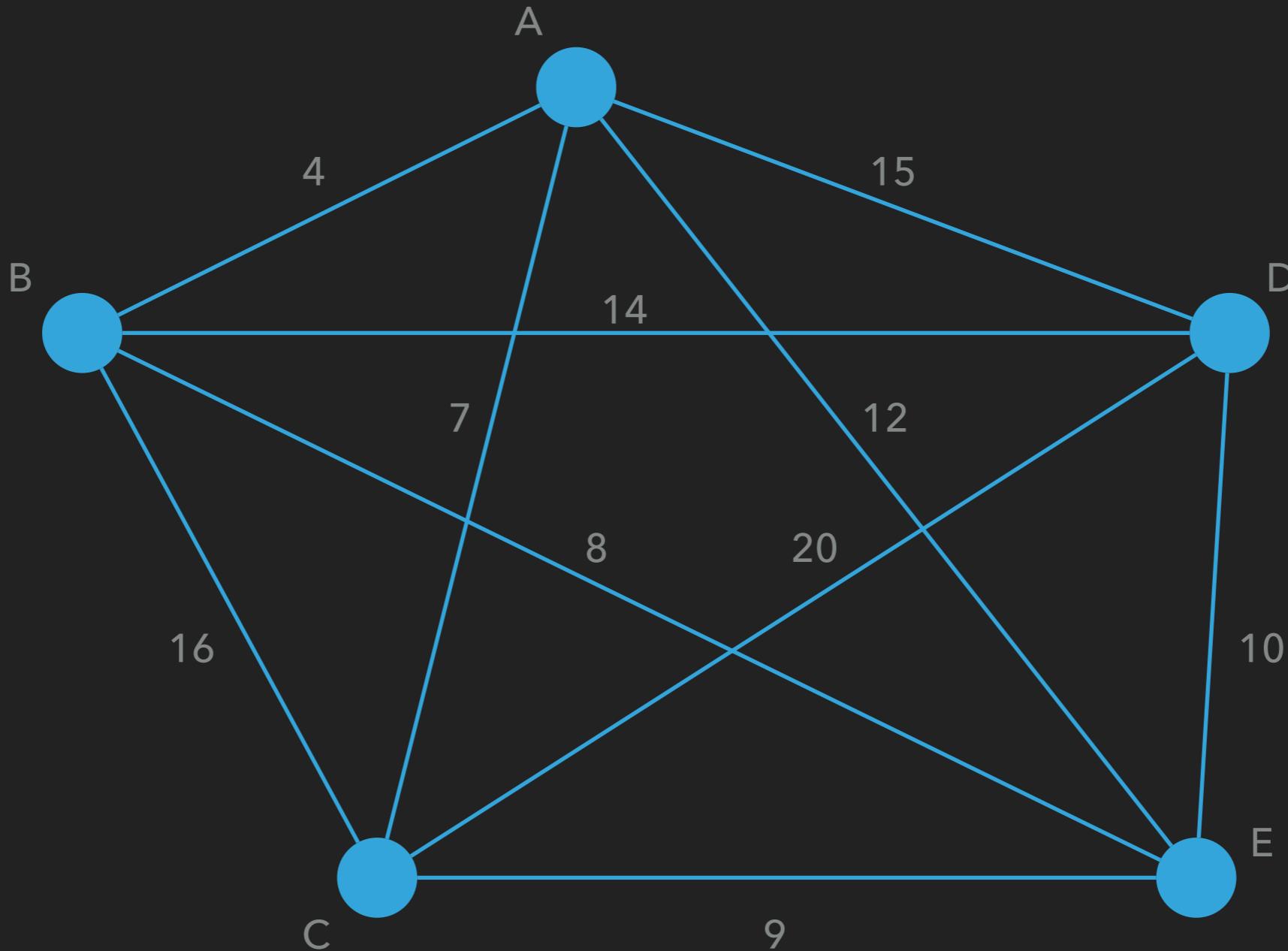
ALGORITMOS DE MELHORIA ITERATIVA

- ▶ **Caminho irrelevante, apenas o estado final importa.**
- ▶ O espaço de busca pode ser num,
 - ▶ conjunto de configurações, onde a melhor satisfaz certas restrições.
 - ▶ conjunto de estados, onde tenta-se melhorar o estados atual.



EXEMPLO
**CAIXEIRO
VIAJANTE**

PROBLEMA DO CAIXEIRO VIAJANTE



Encontrar o menor caminho a ser percorrido que passa por todos os pontos, sendo o inicial o mesmo que o final.

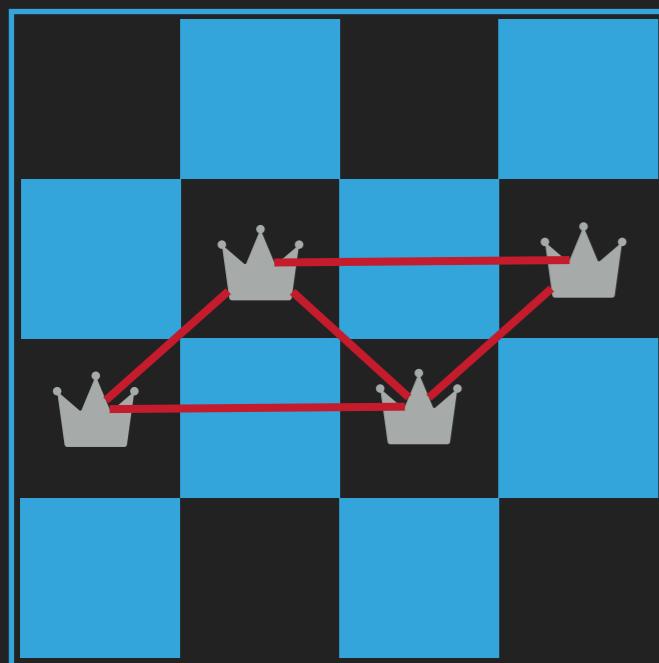


EXEMPLO N RAINHAS

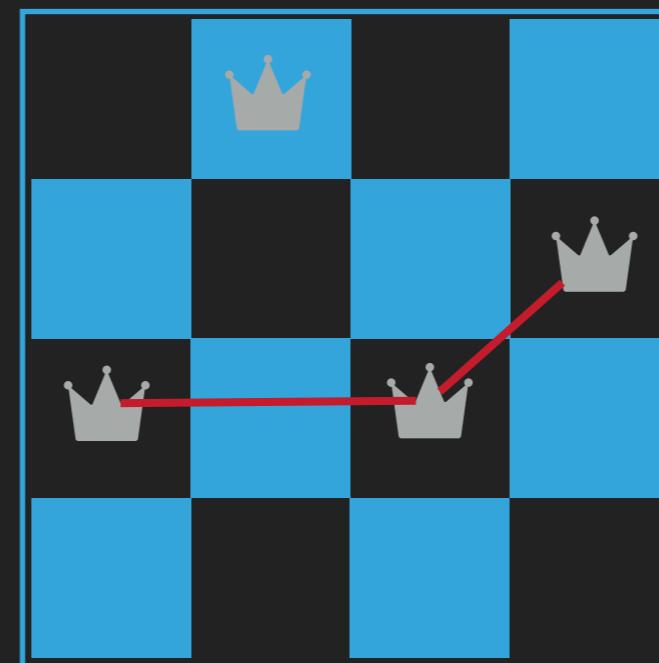
Lucas Baggio Figueira [@lucasfigueira]

PROBLEMA DAS N RAINHAS

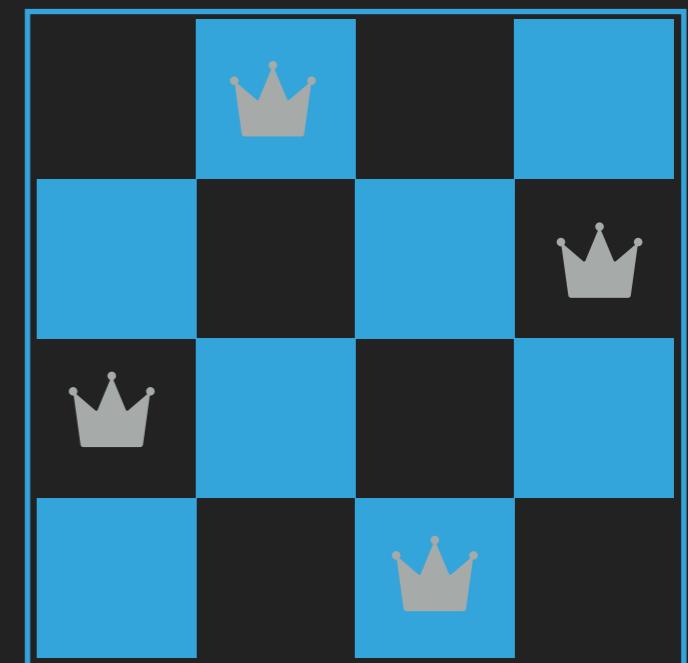
- ▶ Num tabuleiro $n \times n$ posicionar n rainhas de forma que nenhuma esteja sob ataque iminente.



$h = 5$



$h = 2$



$h = 0$



ESTRATÉGIA

HILL CLIMBING
(ESCALADA)

Lucas Baggio Figueira [@lucasfigueira]

```
function Hill-Climbing(problem) returns
    a state that is a local maximum

inputs: problem, a problem

local variables: current, a node
                    neighbour, a node

current  $\leftarrow$  Make-Node(Initial-State[problem])

loop do

    neighbour  $\leftarrow$  a highest-valued successor of current

    if Value[neighbour]  $\leq$  Value[current] then

        return State[current]

    current  $\leftarrow$  neighbour

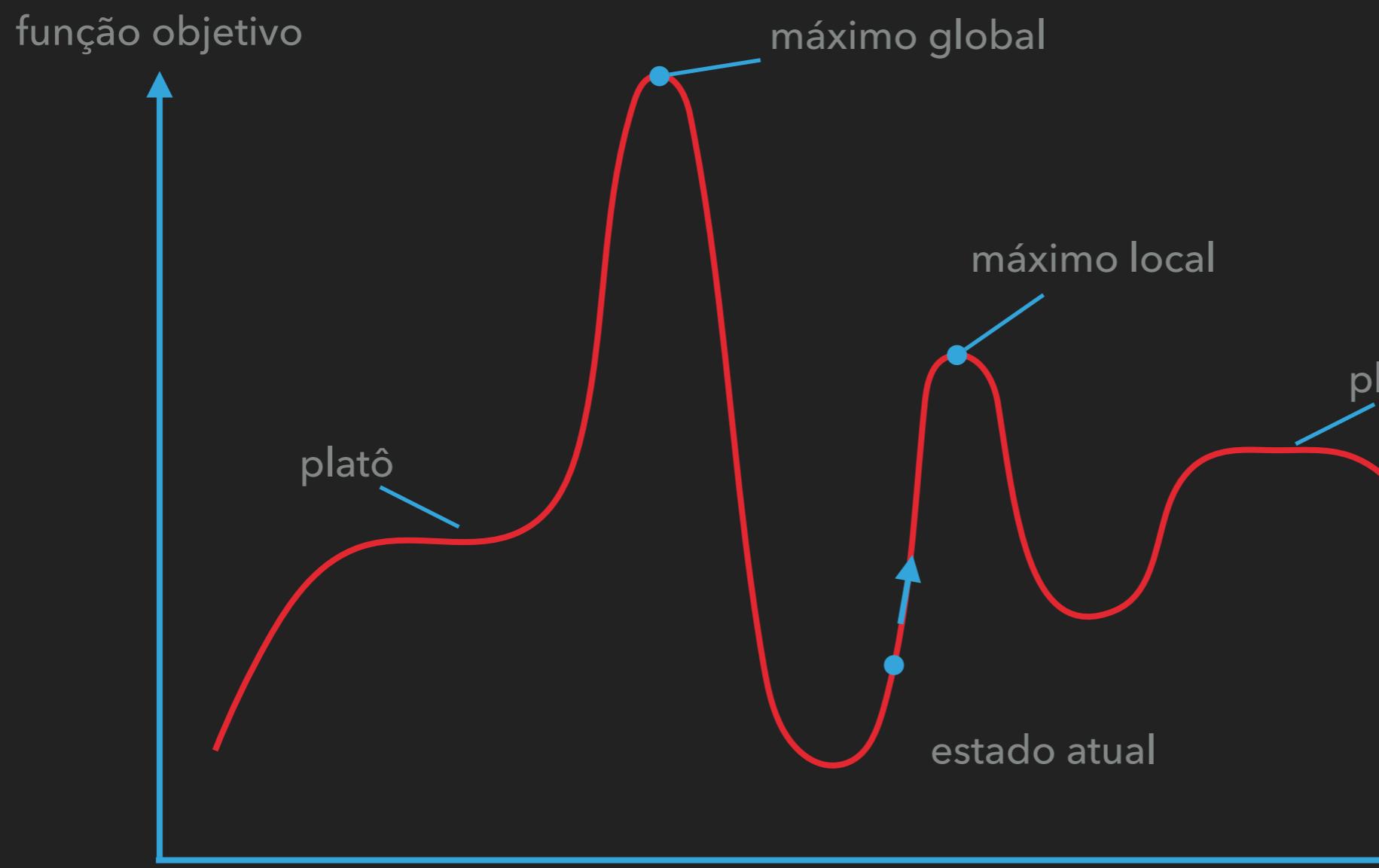
end
```

HILL CLIMBING (ESCALADA)

- Deslocamento do estado atual em direção do máximo (global?)

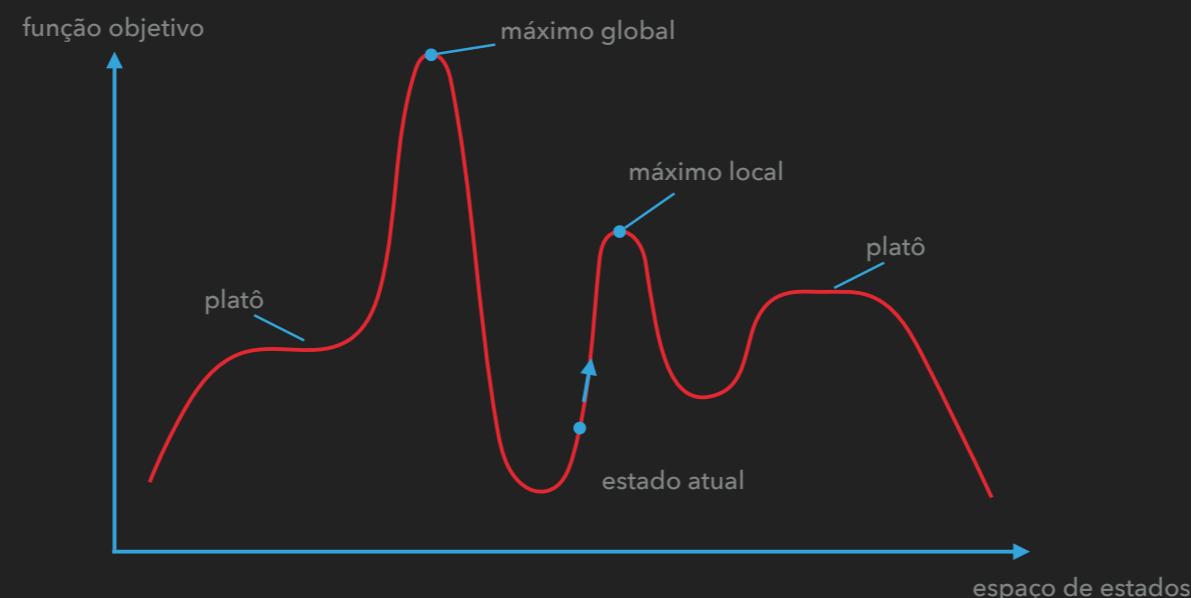
HILL CLIMBING (ESCALADA)

- Deslocamento do estado atual em direção do máximo (global?)



HILL CLIMBING (ESCALADA)

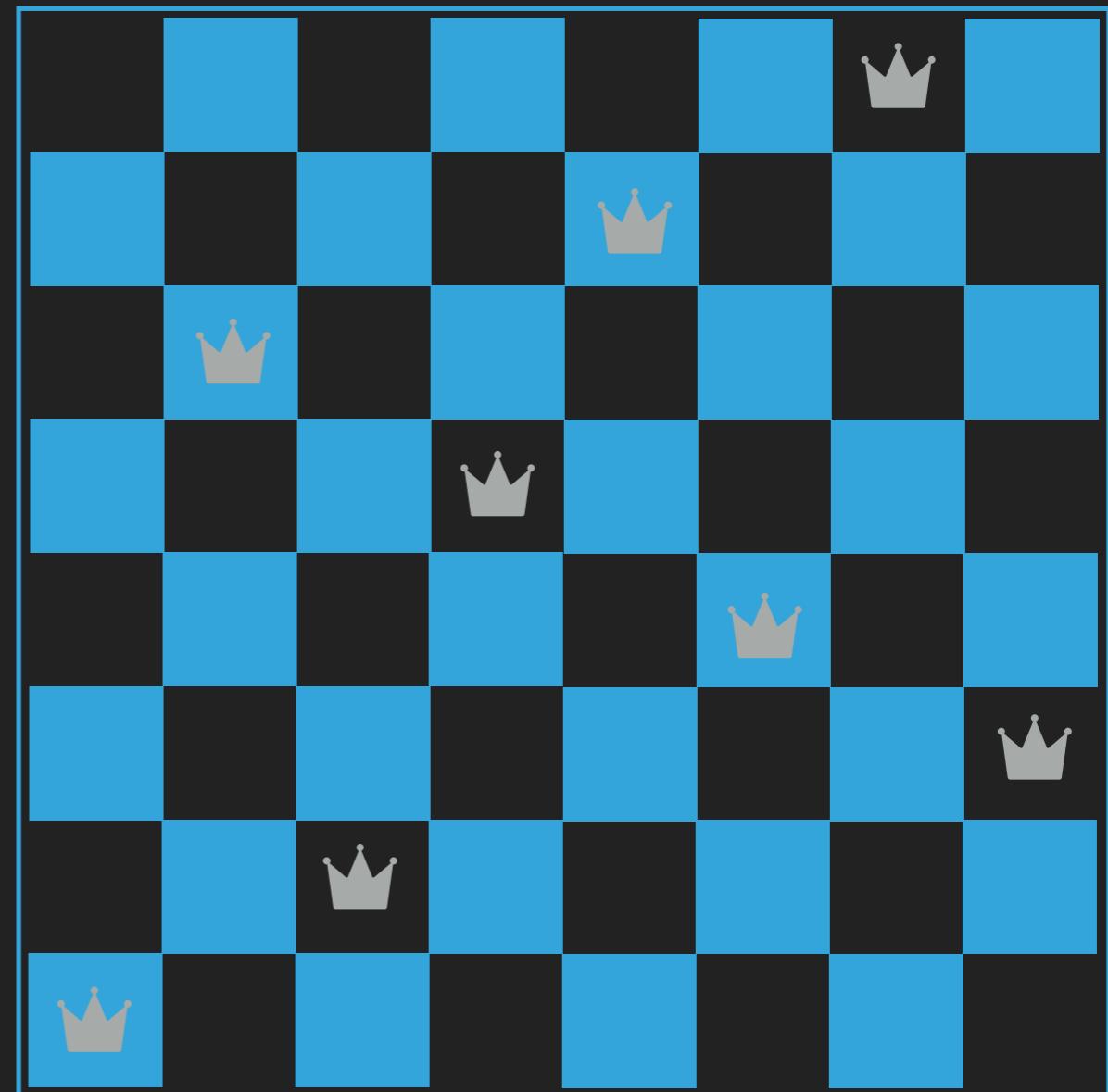
- ▶ Deslocamento do estado atual em direção do máximo (global?)



- ▶ Chamado também de *steepest-descent*
- ▶ Implementação simples
- ▶ Versões determinísticas e estocástico.
- ▶ Problemas NP-Difícil possuem máximos locais em quantidade exponencial.

HILL CLIMBING - 8 RAINHAS

18	12	14	13	13	12	14	14
14	16	13	15	12	14	12	16
14	12	18	13	15	12	14	14
15	14	14	crown	13	16	13	16
crown	14	17	15	crown	14	16	16
17	crown	16	18	15	crown	15	crown
18	14	crown	15	15	14	crown	16
14	14	13	17	12	14	12	18



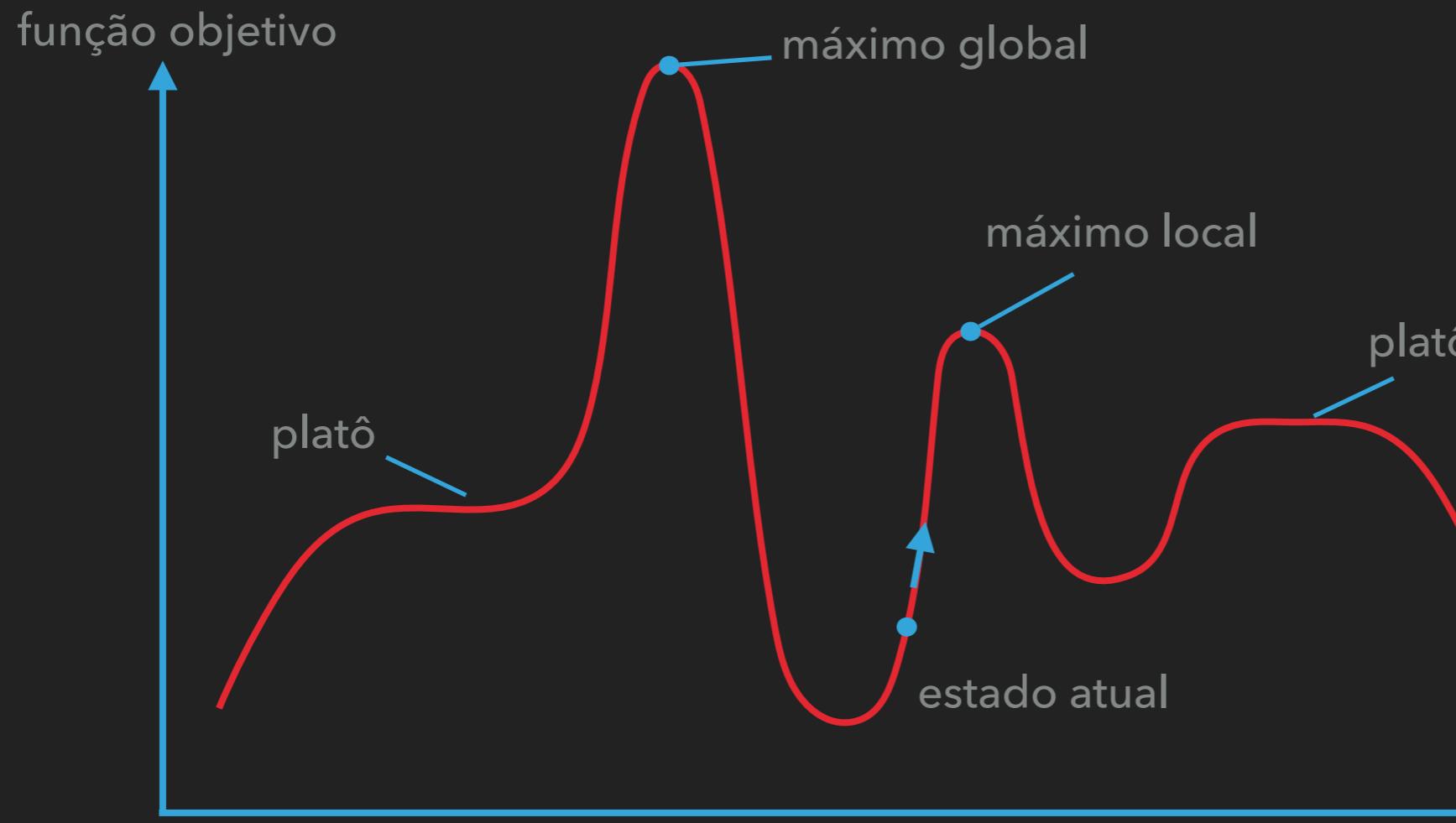


ESTRATÉGIA SIMULATED ANNEALING

Lucas Baggio Figueira [[@lucasfigueira](https://twitter.com/lucasfigueira)]

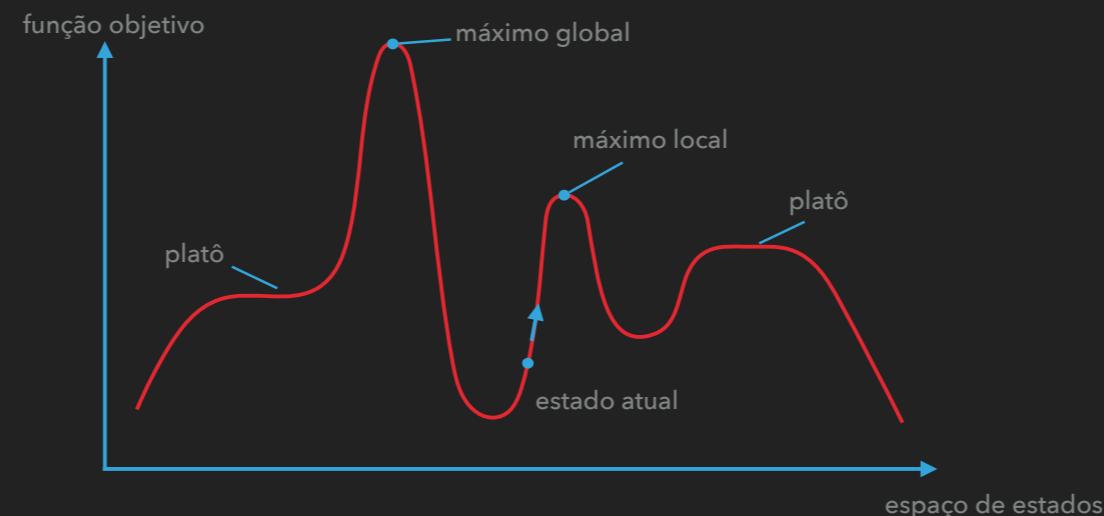
HILL CLIMBING E RANDOM WALKS

- ▶ Nunca “desce a ladeira” - fica preso ao máximo local mais próximo.
- ▶ Vários pontos iniciais (random walk) se mostram ineficiente.



HILL CLIMBING E RANDOM WALKS

- ▶ Nunca “desce a ladeira” - fica preso ao máximo local mais próximo.
- ▶ Vários pontos iniciais (random walk) se mostram ineficiente.



SIMULATED ANNEALING

É O PROCESSO DE COMBINAR CALOR E
FRIO PARA ENDURECER MATERIAIS COMO
VIDRO E AÇO

SIMULATED ANNEALING

SIMULATED ANNEALING

HILL
CLIMBING



RANDOM
WALK

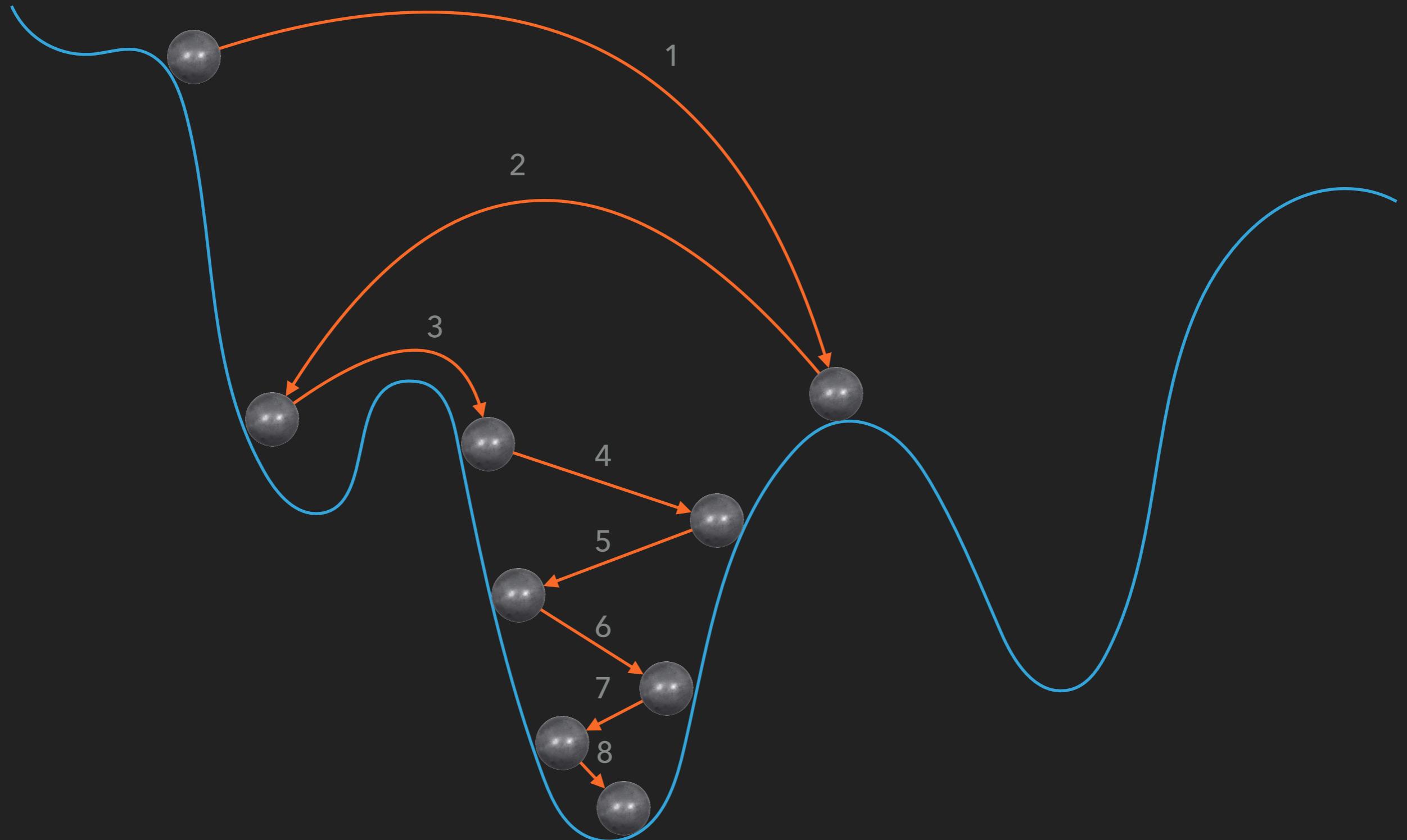
**SIMULATED ANNEALING
PODE SER COMPARADO COM
O GRADIENTE DESCENDENTE**

SIMULATED ANNEALING
PODE SER COMPARADO COM
O GRADIENTE DESCENDENTE

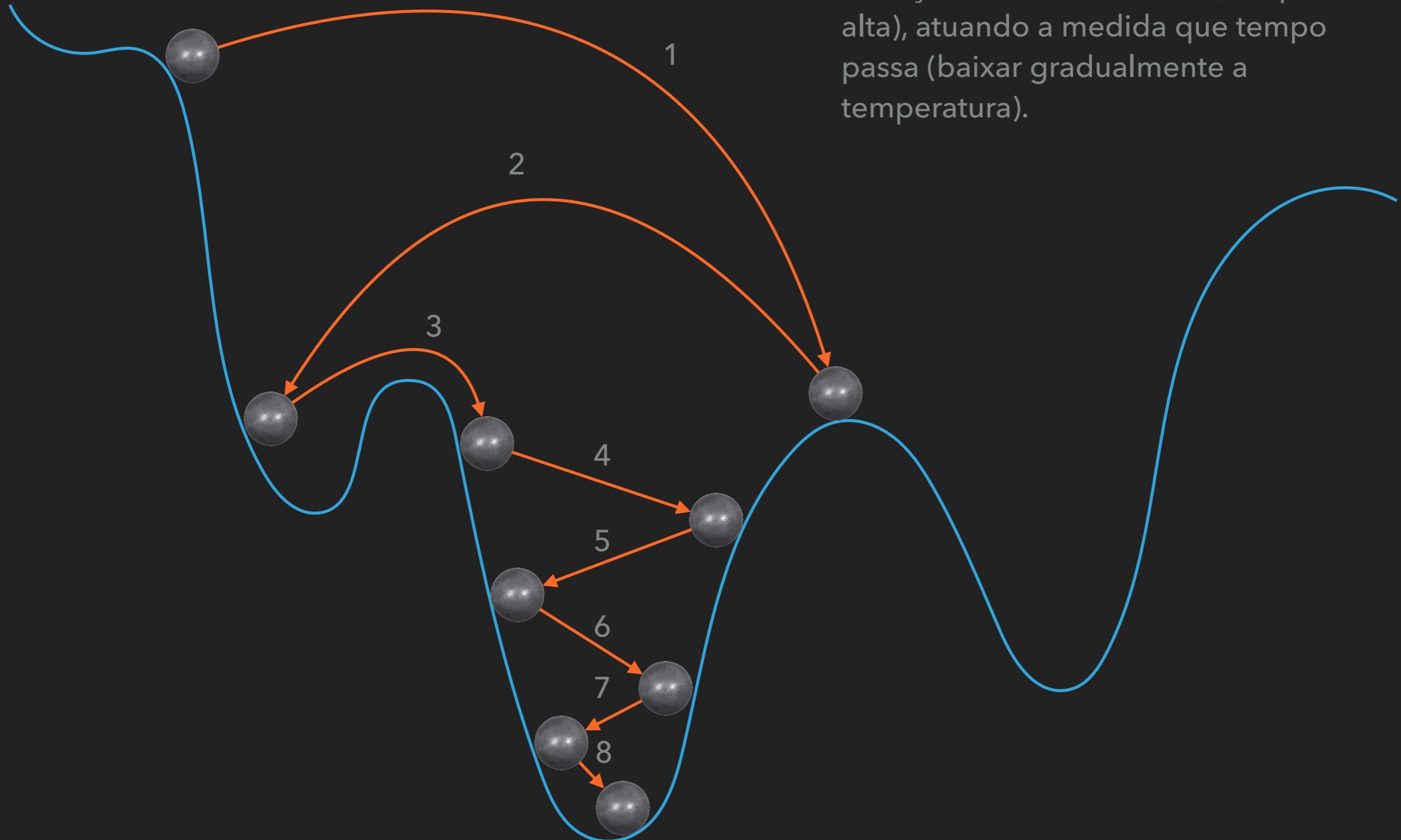


Lucas Baggio Figueira [@lucasfigueira]

GRADIENTE DESCENDENTE



GRADIENTE DESCENDENTE



```

function SIMULATED-ANNEALING(problem,schedule)
    returns a solution state

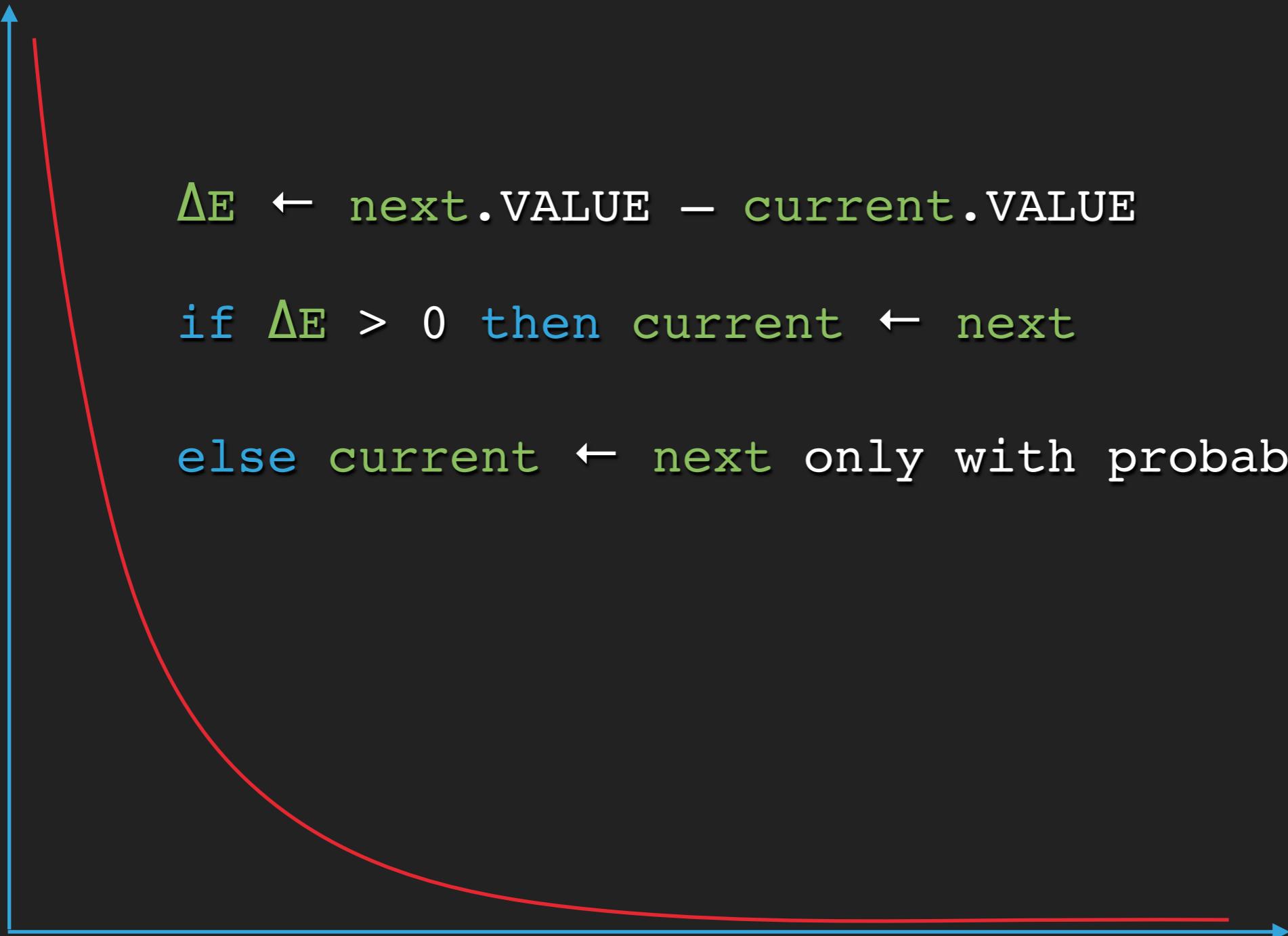
inputs: problem, a problem
        schedule, a mapping from time to "temperature"

current  $\leftarrow$  MAKE-NODE(problem.INITIAL-STATE)

for t = 1 to  $\infty$  do
    T  $\leftarrow$  schedule(t)
    if T = 0 then return current
    next  $\leftarrow$  a randomly selected successor of current
     $\Delta E \leftarrow$  next.VALUE - current.VALUE
    if  $\Delta E > 0$  then current  $\leftarrow$  next
    else current  $\leftarrow$  next only with probability  $e^{\frac{\Delta E}{T}}$ 

```

$$e^{\Delta E/T}$$



```
 $\Delta E \leftarrow next.VALUE - current.VALUE$ 
```

```
if  $\Delta E > 0$  then  $current \leftarrow next$ 
```

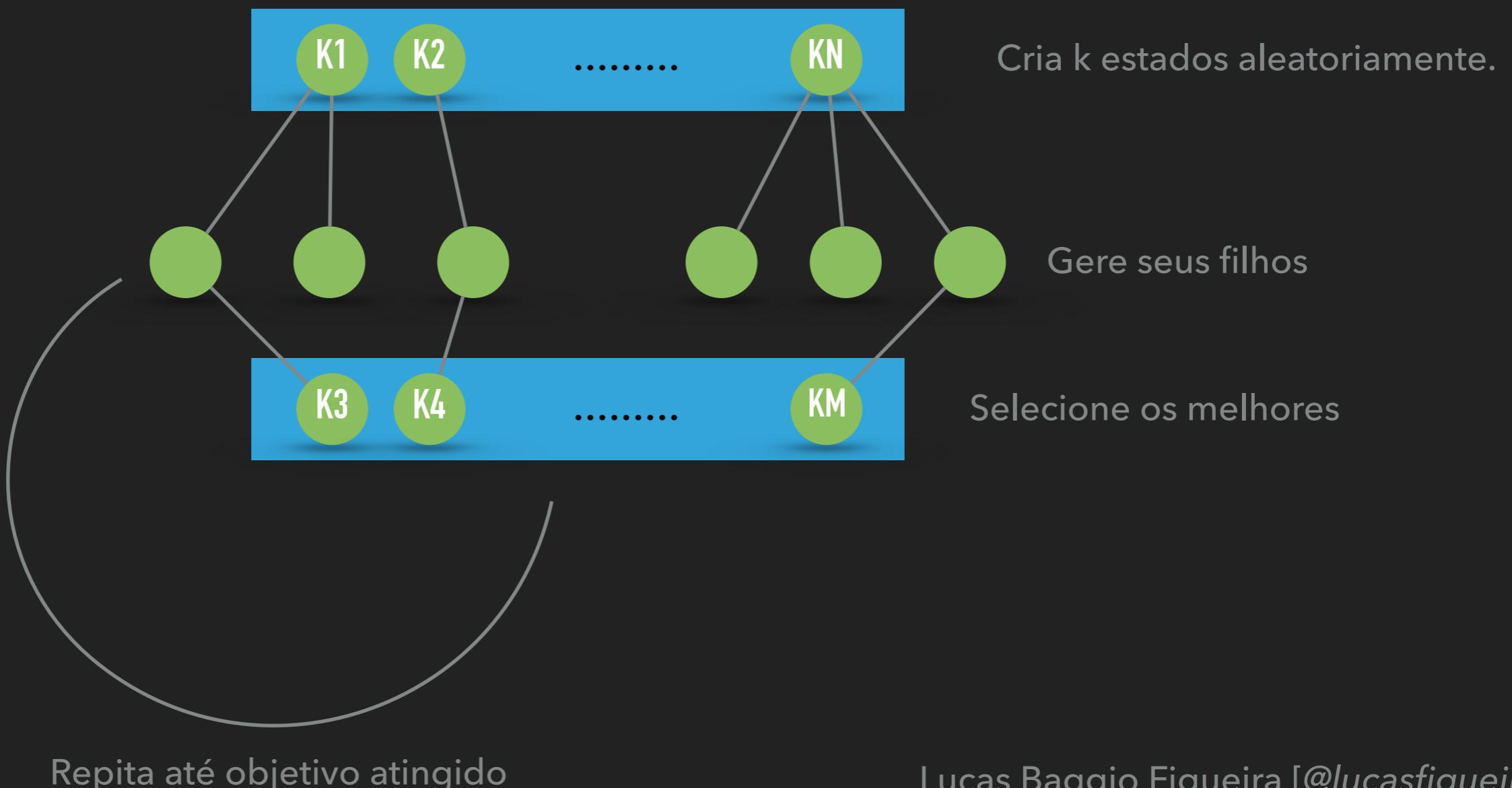
```
else  $current \leftarrow next$  only with probability  $e^{\Delta E/T}$ 
```

LOCAL BEAN SEARCH

- Ao invés de manter apenas um estado, manter e avaliar k estados.

LOCAL BEAN SEARCH

- Ao invés de manter apenas um estado, manter e avaliar k estados.



LOCAL BEAN SEARCH

- Ao invés de manter apenas um estado, manter e avaliar k estados.



Repita até objetivo atingido

Lucas Baggio Figueira [@lucasfigueira]



BUSCA LOCAL E OTIMIZAÇÃO

ALGORITMOS GENÉTICOS

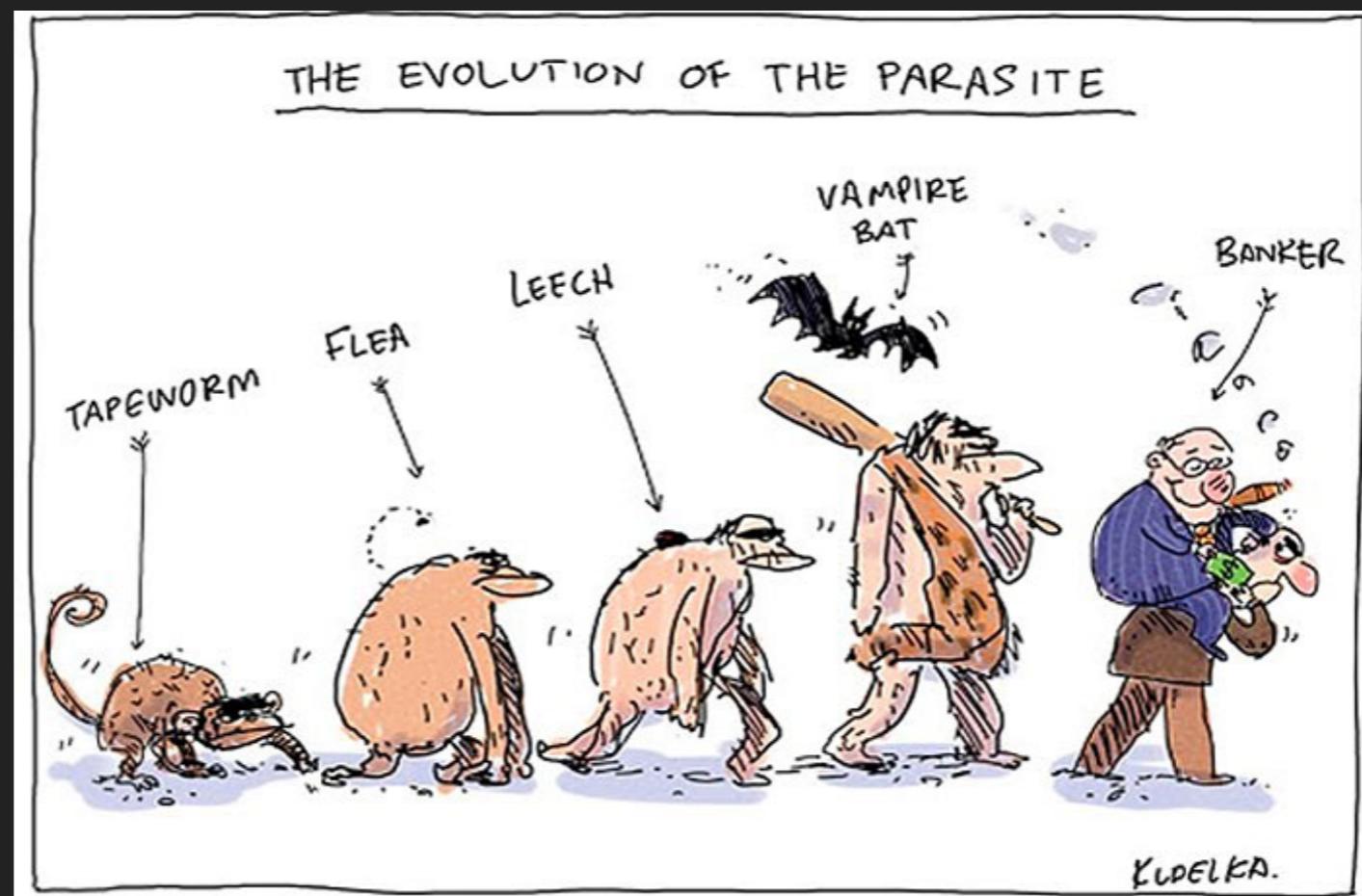
Lucas Baggio Figueira [[@lucasfigueira](#)]

ALGORITMOS GENÉTICOS

- ▶ Variante da *stochastic local beam search*.
- ▶ Analogia com a seleção natural dada sua característica estocástica.

ALGORITMOS GENÉTICOS

- ▶ Variante da *stochastic local beam search*.
- ▶ Analogia com a seleção natural dada sua característica estocástica.

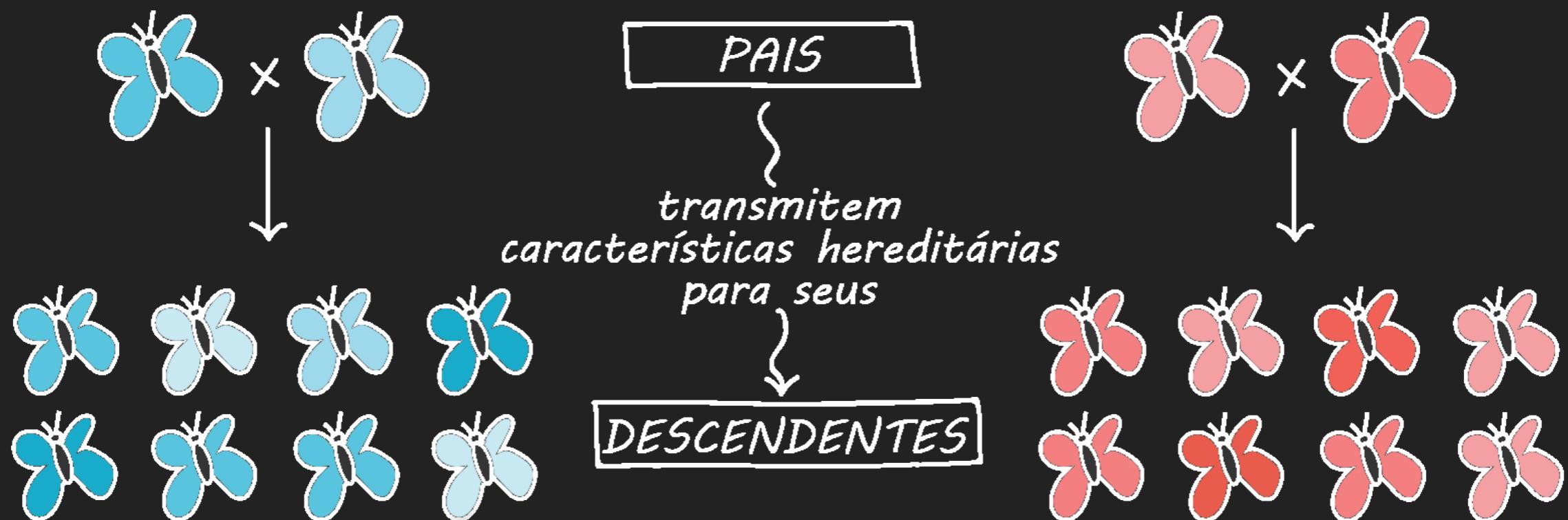


AG - IDÉIA BÁSICA

- ▶ A partir de uma população de estados combina-los em pares para a geração de descendentes.

AG - IDÉIA BÁSICA

- ▶ A partir de uma população de estados combina-los em pares para a geração de descendentes.

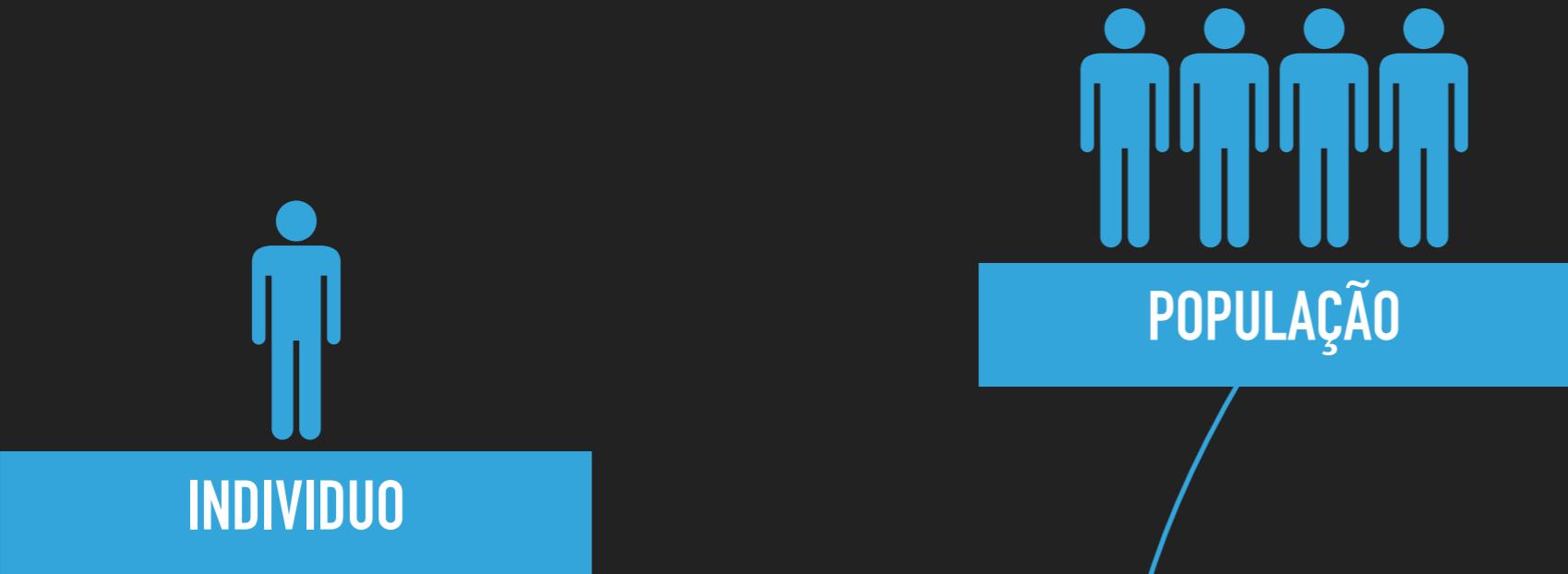


TERMINOLOGIA EM ALGORITMOS GENÉTICOS

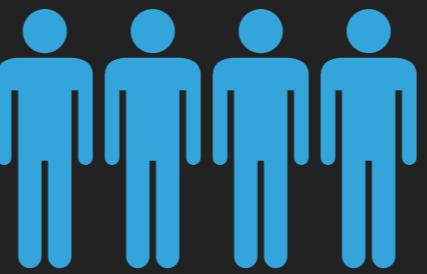


INDIVIDUO

TERMINOLOGIA EM ALGORITMOS GENÉTICOS



TERMINOLOGIA EM ALGORITMOS GENÉTICOS



POPULAÇÃO

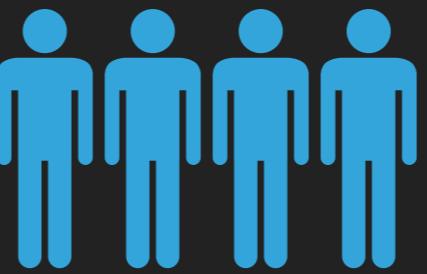


INDIVÍDUO



FUNÇÃO DE FITNESS

TERMINOLOGIA EM ALGORITMOS GENÉTICOS



POPULAÇÃO



INDIVÍDUO

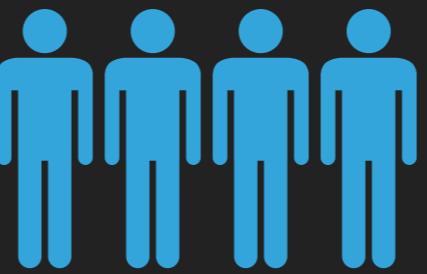


FUNÇÃO DE FITNESS

TERMINOLOGIA EM ALGORITMOS GENÉTICOS

CRUZAMENTO





POPUAÇÃO



INDIVÍDUO



FUNÇÃO DE FITNESS

TERMINOLOGIA EM ALGORITMOS GENÉTICOS

MUTAÇÃO



CRUZAMENTO



```
function GENETIC-ALGORITHM(population, FITNESS-FN)
    returns an individual
inputs: population, a set of individuals
        FITNESS-FN, a function that measures the
                    fitness of an individual
repeat
    new_population ← empty set
    for i = 1 to SIZE(population) do
        x ← RANDOM-SELECTION(population, FITNESS-FN)
        y ← RANDOM-SELECTION(population, FITNESS-FN)
        child ← REPRODUCE(x,y)
        if (small random probability) then
            child ← MUTATE(child)
        add child to new_population
    population ← new_population
until some individual is fit enough,
    or enough time has elapsed
return the best individual in population,
    according to FITNESS-FN
```

```
function REPRODUCE(x,y) returns an individual  
inputs: x , y , parent individuals
```

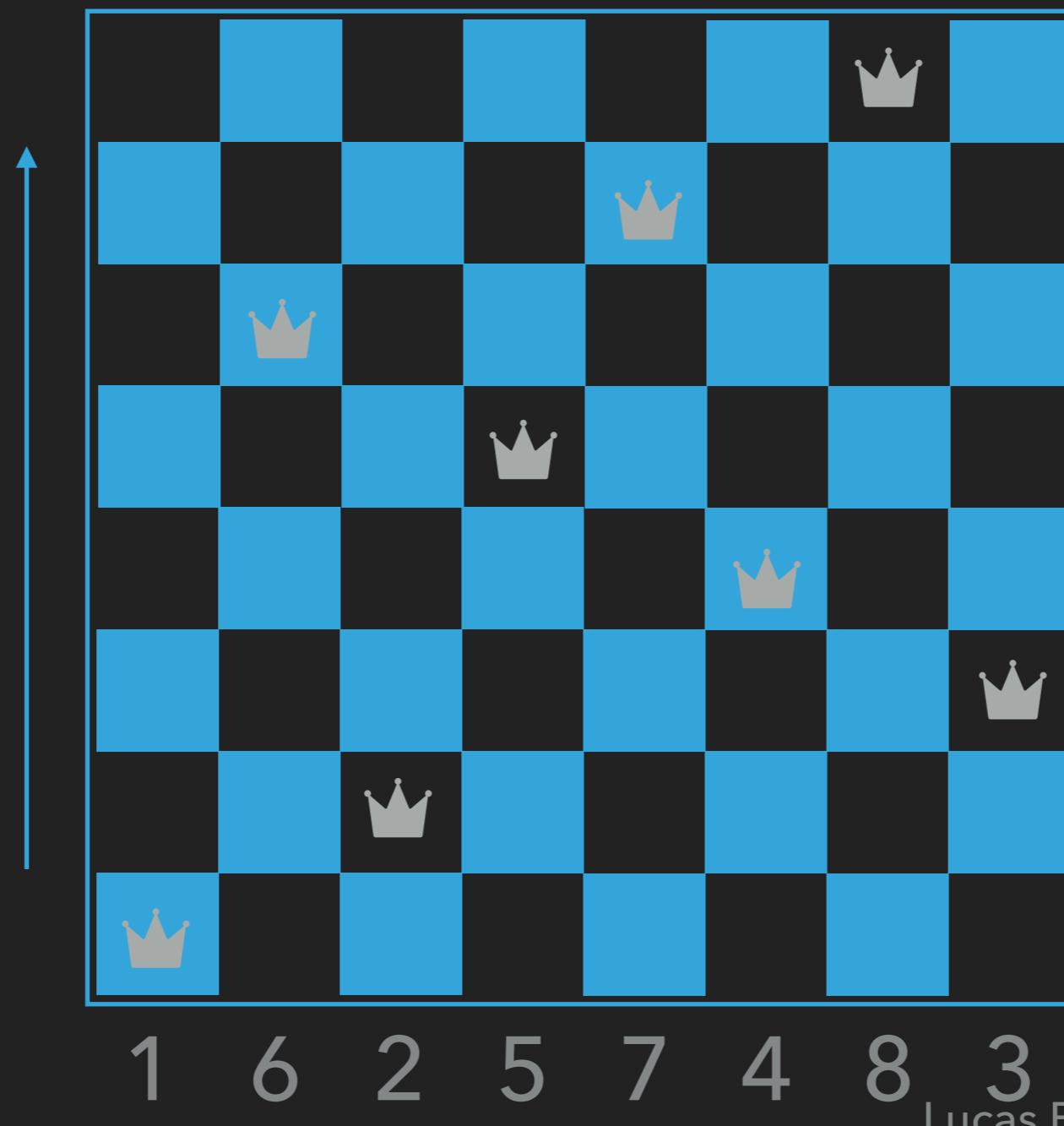
```
n ← LENGTH(x);
```

```
c ← random number from 1 to n
```

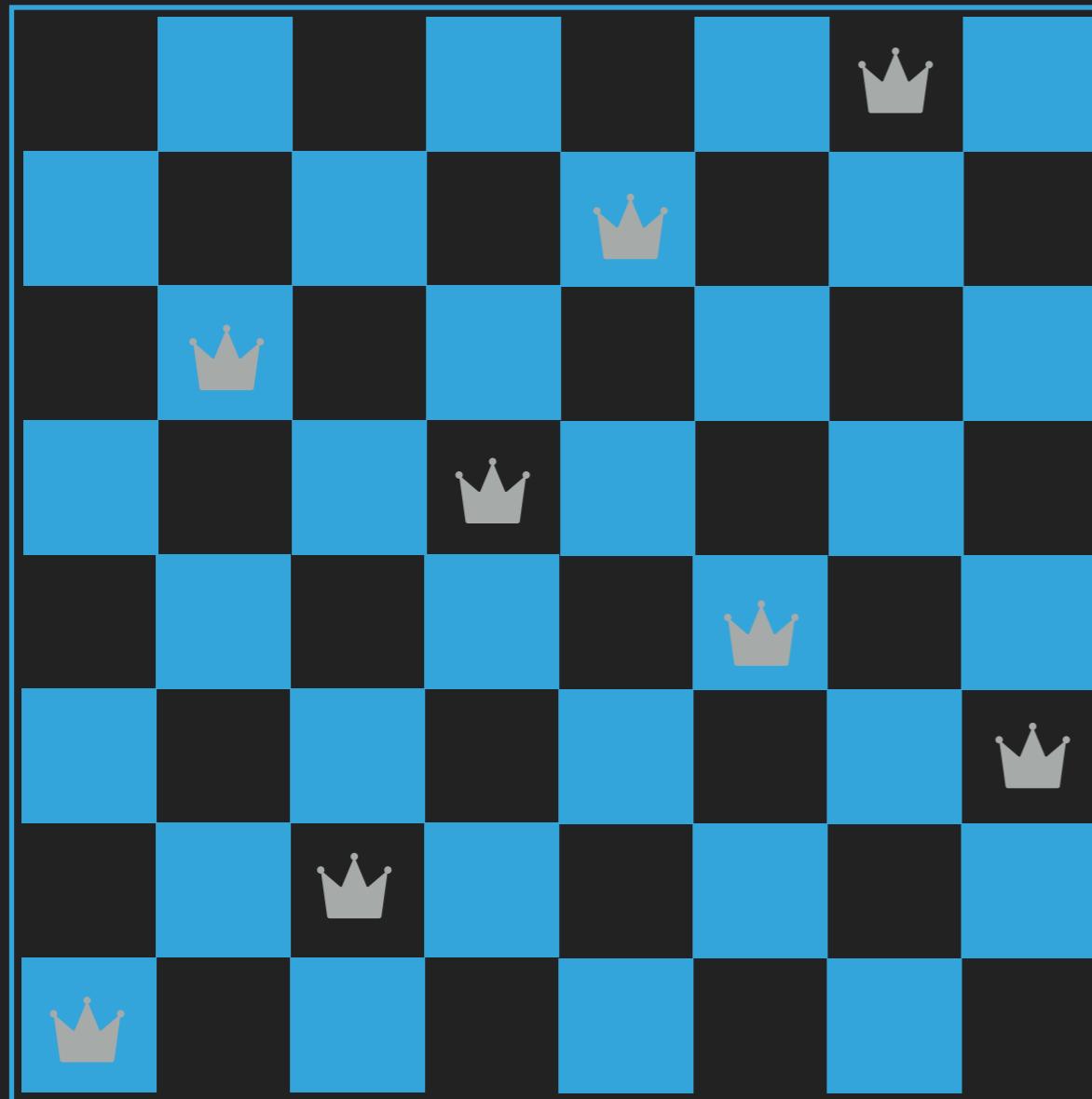
```
return APPEND(SUBSTRING(x,1,c),SUBSTRING(y,c+1,n))
```

ALGORITMOS GENÉTICO NAS 8-RAINHAS

- ▶ Representação do Indivíduo: linha de baixo para cima



ALGORITMOS GENÉTICO NAS 8-RAINHAS



Indivíduo

1	6	2	5	7	4	8	3
---	---	---	---	---	---	---	---

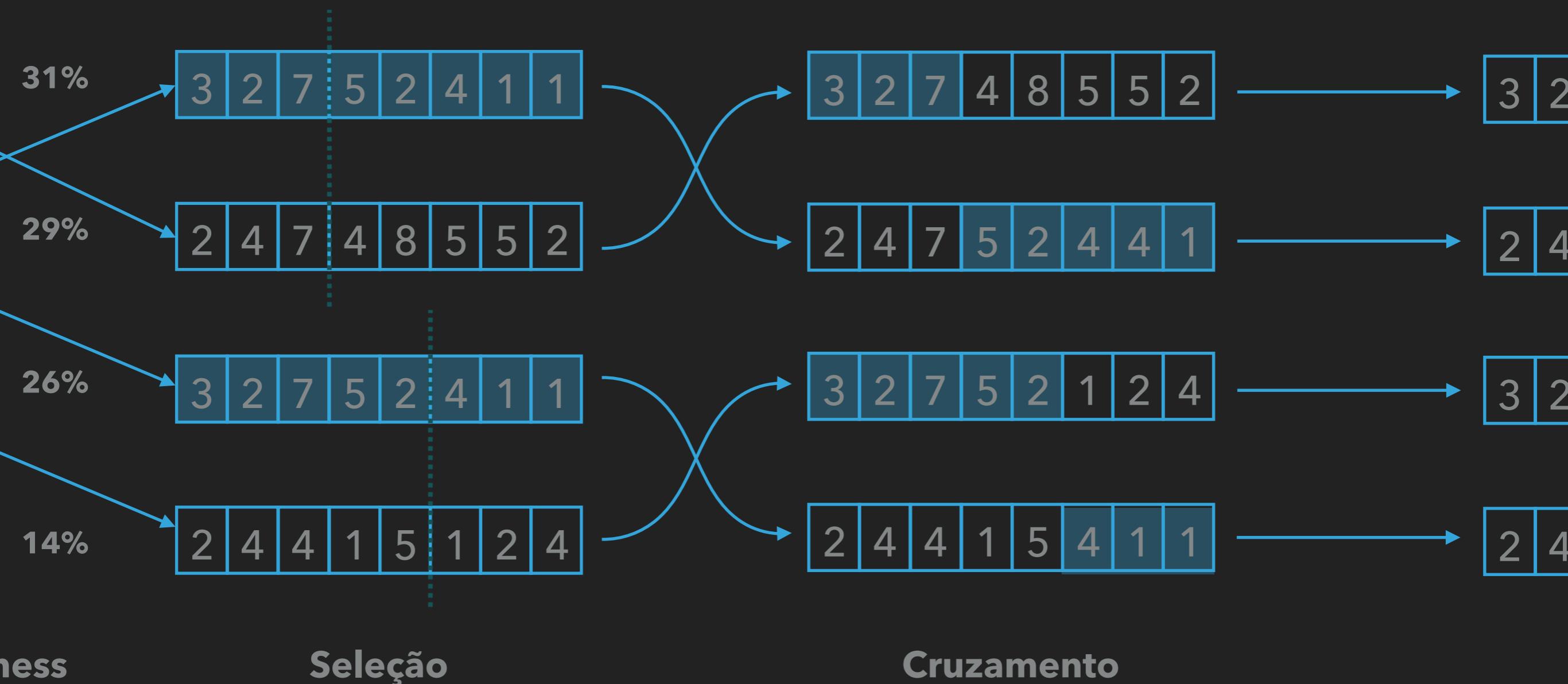
PASSO EVOLUTIVO - 8-RAINHAS

fitness = pares de rainhas que não se atacam



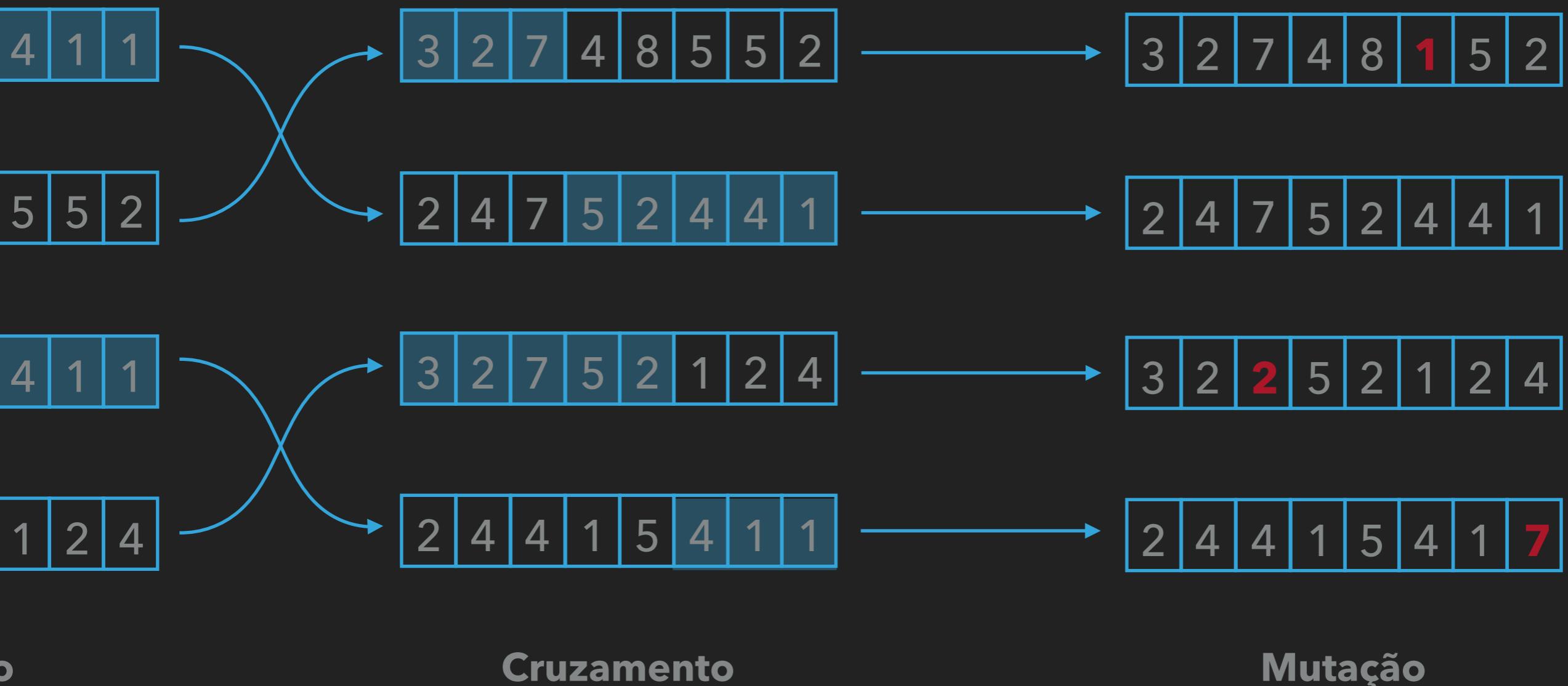
PASSO EVOLUTIVO - 8-RAINHAS

fitness = pares de rainhas que não se atacam



PASSO EVOLUTIVO - 8-RAINHAS

fitness = pares de rainhas que não se atacam



EXERCÍCIOS

1. Usar AG para localizar num espaço de estados com strings de 24 bits aquela composta por bits 1.
2. Usar AG para resolver o problemas das 8-Rainhas.