

In []:

```
--- Algoritmos de Otimização ---

'Subida de encosta'

Algoritmo:

atual <- estado inicial
para t de 1 até inf faça:
    T <- escalonamento(t)
    se t = 0 retorna atual
    próximo <- sucessor qualquer de atual
    delta_E <- valor(próximo) - valor(atual)
    se delta_E > 0 então atual <- próximo
    senão atual <- próximo com probabilidade  $\exp(\text{delta\_E}/T)$ 

'Têmpera simulada (Simulated annealing)'

Algoritmo:

atual <- estado inicial
para t de 1 até inf faça:
    T <- escalonamento(t)
    se t = 0 retorna atual
    próximo <- sucessor qualquer de atual
    delta_E <- valor(próximo) - valor(atual)
    se delta_E > 0 então atual <- próximo
    senão atual <- próximo com probabilidade  $\exp(\text{delta\_E}/T)$ 

Características:
- É completo: seu caráter aleatório garante eventualmente um estado ótimo ser
- O fator  $\exp(\text{delta\_E}/T)$ : a probabilidade de irmos para um estado pior é pequ
```