



IAA - Aula 1

Fábio Nakano



Problema computacional

CLRS seção 1.1



Solução Computacional


No CLRS, há solution, exact solution (p. 1138), optimal solution (p. 1138), approximate solution (p. 1135), feasible solution (p.1128), exact solution(s) (p.1123), different solution(s) (p. 1122), suboptimal solution (p.1111), polynomial-time solution (p.1067), same solution(s) (p.1055), unique solution (p.952), ..., best solution (p.11), efficient solution (p.9), candidate solution (p. 9), compute solution to the problem (p.5)

Aparentemente, “solução” no livro é o resultado da execução de um algoritmo sobre uma entrada específica (instância do problema). A palavra “solução” pode ser caracterizada, por exemplo, eficiente, ótima, candidata, ...



Algoritmo é (também) um método de solução

“Suppose computers were infinitely fast and computer memory was free. Would you have any reason to study algorithms? The answer is yes, if for no other reason than that you would still like to demonstrate that your solution method terminates and does so with the correct answer.” CLRS p.11



Melhor solução, solução “feasible” (fazível??),
heurística, algoritmo aproximado



Exemplo: Problema do caixeiro viajante

Formulação do problema

1. Informal;
2. Suficiente para propor algoritmos computacionais que (tentem) resolvê-lo



Exemplo: Problema do caixeiro viajante

Solução exata: inspecionar todas as rotas possível (geralmente feito por tentativa e erro).

Problema NP (não se sabe se existe algoritmo que resolva o problema em tempo polinomial nos computadores atuais, há esperança (pequena) que possa ser resolvido em tempo polinomial em computadores quânticos;

Heurística mais imediata: ir selecionando sem reconsiderar (guloso)

Algoritmo aproximado: árvore geradora mínima com saltos em grafo euclidiano. O custo ótimo está entre Custo(MST) e $2 * \text{Custo(MST)}$, a complexidade de tempo é $n * \log(n)$