



طراحی الگوریتم (پیچیدگی محاسباتی)



دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان

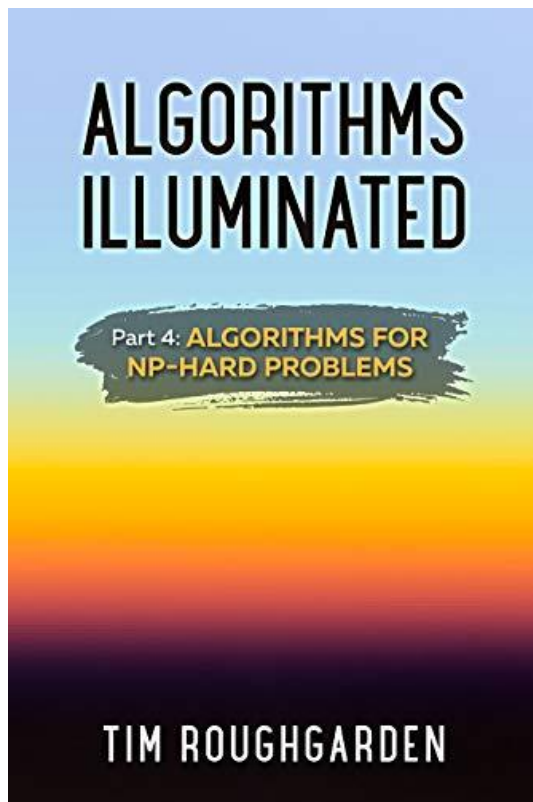
بهار ۹۹



مسائل سخت

تشخیص ☐

راه حل ☐



فصل اول، صفحه ۱



آیا به سادگی می‌توان تشخیص داد یک مساله سخت است؟

درخت پوشای کمینه

ورودی: یک گراف غیرجهت‌دار، و وزن‌های حقیقی برای هر یال.

هدف: یک درخت پوشای کمینه برای گراف ورودی که مجموع وزن یال‌های درخت کمینه باشد.

فروشنده دوره گرد

ورودی: یک گراف غیرجهت‌دار کامل، و وزن‌های حقیقی برای هر یال.

هدف: یک گذر فراگیر برای گراف ورودی که مجموع وزن یال‌های گذر کمینه باشد.



مساله فروشنده دوره گرد

فروشنده دوره گرد

ورودی: یک گراف غیرجهت دار کامل، و وزن های حقیقی برای هر یال.

هدف: یک گذر فراگیر برای گراف ورودی که مجموع وزن یال های درخت کمینه باشد.

Fact

As of this writing (in 2020), there is no known fast algorithm for the TSP.



مساله فروشنده دوره گرد

اعتقاد:

هیچ الگوریتم سریعی برای فروشنده دوره گرد وجود ندارد!

I conjecture that there is no good algorithm for the traveling salesman [sic] problem. My reasons are the same as for any mathematical conjecture: (1) It is a legitimate mathematical possibility, and (2) I do not know.⁶



مسائل سخت و آسان



Problem	Algorithm	Running Time
Sorting	MergeSort	$O(n \log n)$
Strong Components	Kosaraju	$O(m + n)$
Shortest Paths	Dijkstra	$O((m + n) \log n)$
MST	Kruskal	$O((m + n) \log n)$
Sequence Alignment	NW	$O(mn)$
All-Pairs Shortest Paths	Floyd-Warshall	$O(n^3)$



الگوریتم‌های با زمان چند جمله‌ای

Problem	Algorithm	Running Time
Sorting	MergeSort	$O(n \log n)$
Strong Components	Kosaraju	$O(m + n)$
Shortest Paths	Dijkstra	$O((m + n) \log n)$
MST	Kruskal	$O((m + n) \log n)$
Sequence Alignment	NW	$O(mn)$
All-Pairs Shortest Paths	Floyd-Warshall	$O(n^3)$



Courage, Definitions, and Edge Cases

The identification of “easy” with “polynomial-time solvable” is imperfect; a problem might be solved in theory (by an algorithm that technically runs in polynomial time) but not in reality (by an empirically fast algorithm), or vice versa. Anyone with the guts to write down a precise mathematical definition (like polynomial-time solvability) to express a messy real-world concept (like “easy to solve via computer in the physical world”) must be ready for friction between the binary nature of the definition and the fuzziness of reality. The definition will inevitably include or exclude some edge cases that you wish had gone the other way, but this is no excuse to ignore or dismiss a good definition. Polynomial-time solvability has been unreasonably effective at classifying problems as “easy” or “hard” in a way that accords with empirical experience. With a half-century of evidence behind us, we can confidently say that natural polynomial-time solvable problems typically can be solved with practical general-purpose algorithms, and that problems believed to not be polynomial-time solvable typically require significantly more work and domain expertise.

مسائل سخت و آسان



مسائل سخت



مسائل سخت



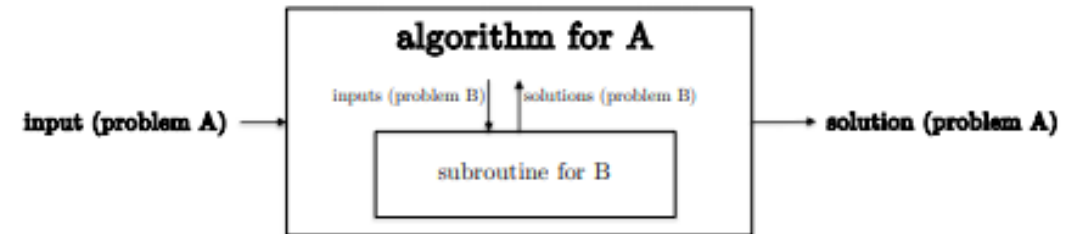
کاهش دادن یک مساله



On Reductions

A problem A *reduces* to a problem B if an algorithm that solves B can be easily translated into one that solves A . For example, the problem of computing the median element of an array reduces to the problem of sorting the array. Reductions are one of the most important concepts in the study of algorithms and their limitations, and they can also have great practical utility.

You should always be on the lookout for reductions. Whenever you encounter a seemingly new problem, always ask: Is the problem a disguised version of one you already know how to solve? Alternatively, can you reduce the general version of the problem to a special case?





سختی مسائل



مسائل سخت



چگونگی تشخیص مسائل سخت



چگونگی تشخیص مسائل سخت (مثال)

ورودی: یک گراف جهت‌دار، وزن یال‌ها، راس شروع

هدف: برای هر راس کوتاهترین مسیر از راس شروع که راس تکراری ندارد.

ورودی: یک گراف جهت‌دار، راس شروع و راس پایانی

هدف: آیا کوتاهترین مسیر از راس شروع به راس پایانی بدون راس تکراری داریم؟



چگونگی تشخیص مسائل سخت (مثال)

ورودی: یک گراف جهت‌دار، وزن یال‌ها، راس شروع

هدف: برای هر راس کوتاهترین مسیر از راس شروع که راس تکراری ندارد.

ورودی: یک گراف جهت‌دار، راس شروع و راس پایانی

هدف: آیا کوتاهترین مسیر از راس شروع به راس پایانی که هر راس دقیقا یکبار دیده شود، داریم؟



Rookie Mistake #5

Devising a reduction in the wrong direction.

اشتباهات متداول

Rookie Mistake #1

Thinking that “NP” stands for “not polynomial.”

Rookie Mistake #2

Saying that a problem is “an NP problem” or “in NP” instead of “NP-hard.”

Rookie Mistake #3

Thinking that NP-hardness doesn’t matter because NP-hard problems can generally be solved in practice.

Rookie Mistake #4

Thinking that advances in computer technology will rescue us from NP-hardness.