canonical problems list

Asked 16 years, 3 months ago Modified 15 years, 6 months ago

Viewed 899 times

5

Does anyone known of a a good reference for canonical CS problems?

I'm thinking of things like "the sorting problem", "the bin packing problem", "the travailing salesman problem" and what not.

edit: websites preferred

reference theory knapsack-problem

Share

Improve this question

Follow

edited Sep 26, 2008 at 18:24

Hank Gay

71.8k • 36 • 161 • 222

asked Aug 30, 2008 at 0:53

BCS

78.3k • 69 • 194 • 298

6 Answers

Sorted by: Highest score (default)

You can probably find the best in an algorithms textbook like Introduction to Algorithms. Though I've never read

4

that particular book, it's quite renowned for being thorough and would probably contain most of the problems you're likely to encounter.

Share Improve this answer

Follow

answered Aug 30, 2008 at 1:00

Kyle Cronin

79k ● 45 ● 151 ● 167

4

"Computers and Intractability: A guide to the theory of NP-Completeness" by Garey and Johnson is a great reference for this sort of thing, although the "solved" problems (in P) are obviously not given much attention in the book.

I'm not aware of any good on-line resources, but Karp's seminal paper Reducibility among Combinatorial Problems (1972) on reductions and complexity is probably the "canonical" reference for Hard Problems.

Share Improve this answer

Follow

edited Jun 19, 2009 at 7:43

RBerteig

43.3k ● 7 ● 90 ● 130

answered Aug 30, 2008 at 1:07

rcreswick

16.8k ● 15 ● 60 ● 70

3

Have you looked at Wikipedia's Category:Computational problems and Category:NP Complete Problems pages?

It's probably not complete, but they look like good starting points. Wikipedia seems to do pretty well in CS topics.

Share Improve this answer

Follow

answered Aug 30, 2008 at 3:14

cpm

1,543 ● 10 ● 16

3

I don't think you'll find the answers to all those problems in only one book. I've never seen any decent, comprehensive website on algorithms, so I'd recommend you to stick to the books. That said, you can always get some introductory material on canonical algorithm texts (there are always three I usually recommend: CLRS, Manber, Aho, Hopcroft and Ullman (this one is a bit out of date in some key topics, but it's so formal and well-written that it's a must-read). All of them contain important combinatorial problems that are, in some sense, canonical problems in computer science. After learning some fundamentals in graph theory you'll be able to move to Network Flows and Linear Programming. These comprise a set of techniques that will ultimately solve most problems you'll encounter (linear programming with the variables restricted to integer values is NP-hard). Network flows deals with problems defined on graphs (with weighted/capacitated edges) with very interesting applications in fields that seemingly have no relationship to graph theory whatsoever. THE textbook on this is Ahuja, Magnanti and Orlin's. Linear programming is some kind of superset of network flows, and deals with optimizing a linear function on variables subject to

restrictions in the form of a linear system of equations. A book that emphasizes the relationship to network flows is Bazaraa's. Then you can move on to integer programming, a very valuable tool that presents many natural techniques for modelling problems like bin packing, task scheduling, the knapsack problem, and so on. A good reference would be L. Wolsey's book.

Share Improve this answer

Follow

answered Sep 17, 2008 at 23:26

Leandro

161 ● 2 ● 8

Note: I'm more looking for the list of Problems than the list of Solutions. Good answer anyway. – BCS Sep 18, 2008 at 20:43

2

You **definitely** want to look at NIST's Dictionary of Algorithms and Data Structures. It's got the traveling salesman problem, the Byzantine generals problem, the dining philosophers' problem, the knapsack problem (= your "bin packing problem", I think), the cutting stock problem, the eight queens problem, the knight's tour problem, the busy beaver problem, the halting problem, etc. etc.

It doesn't have the firing squad synchronization problem (I'm surprised about that omission) or the Jeep problem (more logistics than computer science).

Interestingly enough there's a blog on codinghorror.com which talks about some of these in puzzle form. (I can't remember whether I've read Smullyan's book cited in the blog, but he is a good compiler of puzzles & philosophical musings. Martin Gardner and Douglas Hofstadter and H.E. Dudeney are others.)

Also maybe check out the Stony Brook Algorithm Repository.

(Or look up "combinatorial problems" on google, or search for "problem" in Wolfram Mathworld or look at Hilbert's problems, but in all these links many of them are more pure-mathematics than computer science.)

Share Improve this answer

Follow

answered Jan 24, 2009 at 21:49

Jason S

189k ● 171 ● 630 ● 995

bin packing is a like knapsack but you put everything in and minimize the bin count – BCS Jan 25, 2009 at 0:52

0

@rcreswick those sound like good references but fall a bit shy of what I'm thinking of. (However, for all I know, it's the best there is)

I'm going to not mark anything as accepted in hopes people might find a better reference.

Meanwhile, I'm going to list a few problems here, fell free to add more

The sorting problem Find an order for a set that is monotonic in a given way

The bin packing problem partition a set into a minimum number of sets where each subset is "smaller" than some limit

The travailing salesman problem Find a Hamiltonian cycle in a weighted graph with the minimum total weight

Share Improve this answer Follow

edited Jan 9, 2009 at 21:30

answered Sep 1, 2008 at 4:27

BCS

78.3k ● 69 ● 194 ● 298