1.1 Algorithms

keywords

algorithm, computational procedure, input, output, problem, sorting, instance of a problem, correct, solving, data structure, NP-complete problems, efficiency, good enough, parallelism,

Summary

An algorithm is a series of logical steps taken upon a piece of information, an input, that produces a result, or output, based on the constraints and requirements of said steps and the properties of the information given.

They are used mostly for problem solving and are the basis of computational procedure, where instructions are followed by a system or an individual to solve it. These instructions are presented in a step by step basis where a certain result from a certain input is expected and answers, or steps, are applied for each result and part of the proccess.

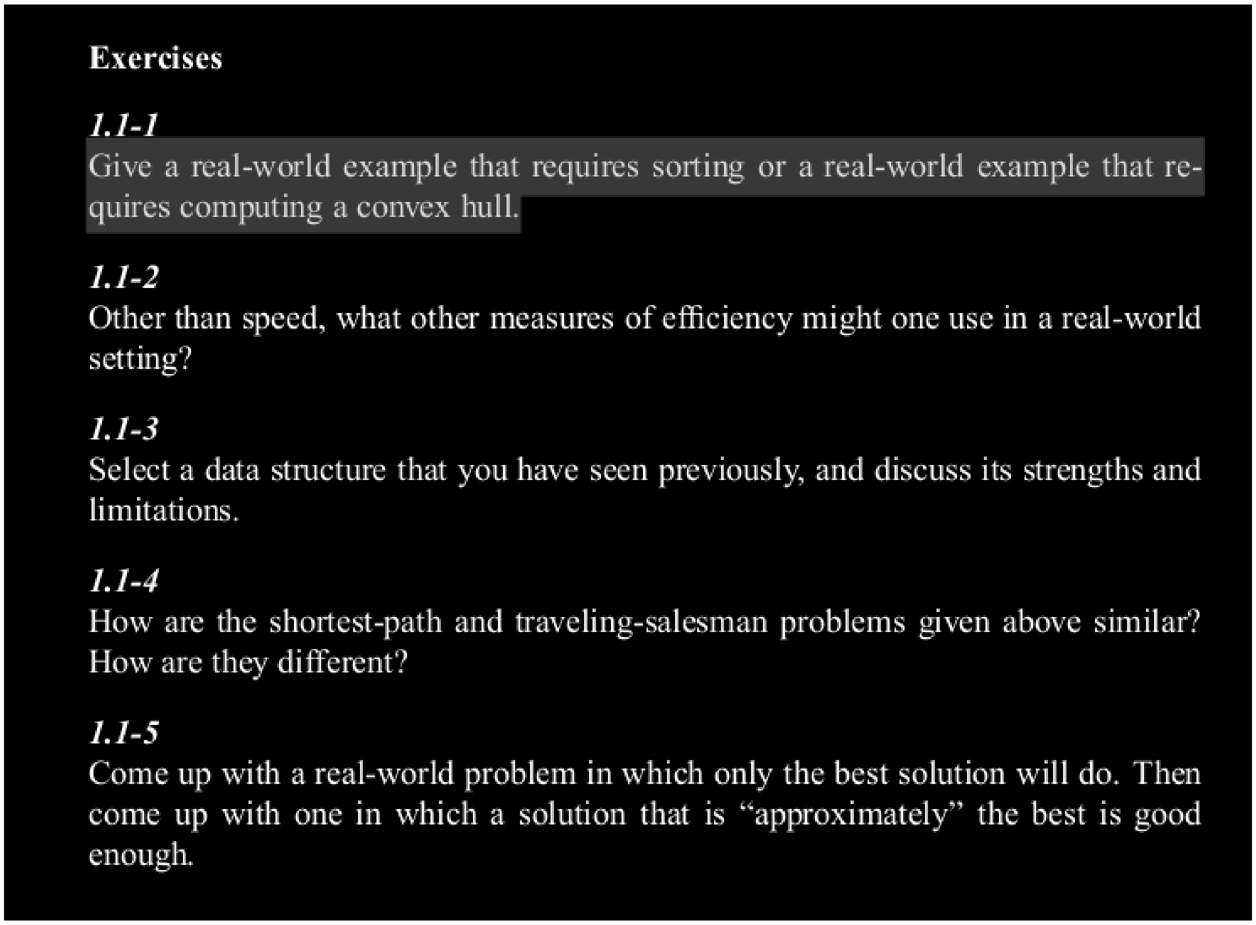
Algorithms can be applied to Data Structures in order to sort them, make them more efficient or simple find a solution that is good enough for a problem. There are algorithmic problems that currently lack a 'best' solution, such as the travelling salesman problem, but this doesn't mean that one doesn't exist - just that it hasn't been found.

Knowing that these types of problems exist are not an incentive for the adventuring problem-solver to toil away in search of their optimal solution, but instead learning to recognize them can prevent hours wasted attempting to solve the unsolvable.

Sometimes the best solution to a problem is one that is close to the optimal but not quite there, just good enough.

Exercises

Answering this without knowing anything about the topic.


Exercises

1.1-1
Give a real-world example that requires sorting or a real-world example that requires computing a convex hull.

1.1-2
Other than speed, what other measures of efficiency might one use in a real-world setting?

1.1-3
Select a data structure that you have seen previously, and discuss its strengths and limitations.

1.1-4
How are the shortest-path and traveling-salesman problems given above similar? How are they different?

1.1-5
Come up with a real-world problem in which only the best solution will do. Then come up with one in which a solution that is "approximately" the best is good enough.

1. requires sorting: food shipments in a store are kept in a data base with their expiration date attatched. These dates vary from days to months from a product to another, A sorting algorithm would present them in order, from the ones that would expire the soonest to the ones that would the latest. This would prevent that the store kept old food from being offered in time.
2. Convenience: easy to use and understand.
   Clean: details that won't be used aren't shown.
3. Array. Simple and clear, the array allows the programmer to store almost anything and access it with an indexation system. It can be naturally sorted and mapped with functions, producing outputs with valuable and curated information.
   Weakness: Array's simple organization means that trying to sort through it and find an item often includes examining the enterity of it until the element is found.
4. They are similar: both require the solver to find the fastest and least resource intensive choice to a problem.

They are different: at this time I don't know how they are different.

5. Choose where I am going to buy my groceries.
   Best solution: the best place will be the
   - shortest to my home
   - lowest prices
   - least amount of waiting in lines or commuting inside the shop
   Aprox good solution:
   - One that fullfills 2 out of the 3 conditions. (A shop is expensive but It's crossing the street and barely has any lines).
   - One whose only good condition is so good it trumps the bad quality of the other 2 (Example: is far away and has lines, but item cost is 25% of the original).
   - One with a decent average of all the conditions
   — Shop A: medium distance, very expensive, few lines.
   — Shop B: short distance, medium prices, long lines.
   — Shop C: medium distance, medium prices, medium lines.
   Shop C would be OK. The others have their strengths and weaknesses but the most average would work the best for most buyers.