

Developing Algorithms

Which of the following are true statements? Choose all that apply.

- Reasonable-looking algorithms tend to be correct. not necessarily – can't use "it looks plausible" as a substitute for a careful argument about correctness
- "Find the best route between two cities" is a well-defined task. false – what is "best"?
- Alice has been tasked with finding the fastest route between cities A and B that avoids interstate highways. This compound goal will be difficult to reason about in order to prove her algorithm correct.
- Bob has been tasked with finding the fastest route between cities A and B with the lowest total cost in tolls. This compound goal will be difficult to reason about in order to prove his algorithm correct. true – fastest and cheapest may be at odds with each other
- If your idea is not clearly revealed when you express an algorithm, then you are using too low-level a notation to describe it. true – details obscure ideas

false – as a shortest path task, can easily just remove (or ignore) interstate edges (no conflict between shortest and not interstate)

"difficult to reason about" includes conflicting goals and things not concrete enough to be able to deduce enough to prove correctness

Correctness

Establishing correctness is a key element of designing algorithms.

- that an algorithm is correct is often not as obvious as it is for data structures, especially for optimization problems or when cleverness is needed to improve running times

Reasoning about correctness needs more than just "it looks reasonable".

- proof involves a chain of reasoning from assumptions to end result

Correctness

Ingredients.

- a clear problem specification is essential
 - defines the set of allowed input instances
 - defines the required properties for the output

Impediments.

- having too broad a class of input instances
 - may need a more restricted problem in order to find an efficient algorithm
- a poorly defined question
 - e.g. looking for the "best" solution without defining what that is
- compound goals that can't be achieved simultaneously or which become difficult to reason about
- wrong level of detail in the algorithm
 - too much detail obscures the idea
 - not enough detail means you don't have enough to work with