Core Objectives for Lesson 1:

- a general understanding that some problems cannot be solved with algorithms at all (and student should at least remember that the Halting Problem is an example)

- a general understanding that some solvable problems are not feasibly solvable and that SubsetSum is an example

- a general understanding (from the exercises) of "greedy strategies" for algorithm design

- a general understanding that sometimes hard problems have the characteristic that, though finding a solution is not known to be feasible, checking a solution is feasible; problems whose solutions can be checked in a feasible amount of time are said to belong to NP; student should know that SubsetSum belongs to NP

- SCI: Just as creation emerges from the concentration of unboundedness to a point and subsequent expansion of point to infinity, so likewise the computationally solvable problems in computer science form just a "point value" within the field of all possible problems -- a highly specialized class of problems that admit polynomial time algorithmic solutions.