0) Consider the coin-change problem: Given a set of US coin types \{1, 5, 10, 25, 50\} and an amount of change to be returned (say 29-cents), determine the fewest number of coins for this amount of change.

a) What intuitive algorithm would you use to solve this problem?

b) Does this greedy algorithm always give the fewest number of coins?

**Divide-and-Conquer** Idea: Solve problem by:
- dividing it into small problem(s)
- solving the smaller problem(s)
- use solution(s) to smaller problem(s) to solve original problem

Consider the coin-change problem: Given a set of coin types (say \{1, 5, 10, 12, 25, 50\}) and an amount of change to be returned (say 29-cents), determine the fewest number of coins for this amount of change.

1) What determines the size of the problem?

2) How could we divide the coin-change problem for 29-cents into smaller problems?

3) If we knew the solution to these smaller problems, how would we be able to solve the original problem?
4) After we give back the first coin, which smaller amounts of change do we have?

Original Problem

Possible First Coin

- 1-cent coin
- 5-cent coin
- 10-cent coin
- 25-cent coin
- 50-cent coin

Smaller problems

5) If we knew the fewest number of coins needed for each possible smaller problem, then how could determine the fewest number of coins needed for the original problem?

6) Complete a recursive relationship for the fewest number of coins.

\[
\text{FewestCoins(change)} = \begin{cases} 
\min( \text{FewestCoins(coin \in \text{CoinSet})} ) + 1 & \text{if change} \not\in \text{CoinSet} \\
\text{if change} \in \text{CoinSet} 
\end{cases}
\]

7) In a dynamic programming coin-change solution, you fill an array fewestCoins from 0 to the amount of change.

An element of fewestCoins stores the fewest number of coins necessary for the amount of change corresponding to its index value. For 29-cents using the set of coin types \{1, 5, 10, 12, 25, 50\}, determine the value of fewestCoins[0] to fewestCoins[7].

\[
fewestCoins: \begin{array}{cccccccc}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 29 \\
\end{array}
\]