The time-complexity of Linear Programming

An algorithm runs in time $f(n)$ if on any input of $n$ symbols, it produces an answer in at most $f(n)$ steps.

A class of problems, e.g., linear programming problems, is in the class P of *polynomial time solvable* problems if there is an algorithm which solves the problem in polynomial time, i.e., in time $p(n)$ for some polynomial $p(n)$. Similarly problems are classified as being *linear* or *exponential* if the time required to solve them is linear or exponential in the size of the input.

**Factorization problem.** Given $N \in \{0,1,2, \ldots \}$, find its first factor. Is this a polynomial-time problem?
For many hard problems, solutions can be validated in polynomial-time. Such problems are classified as *NP problems*. $P \subseteq \text{NP}$. Finding a nontrivial factor $q$ of a number $n$ is hard, but it is easy to check a number $q$ is a factor of $n$.

Biggest open problem in computer science: Does $P = \text{NP}$?

Most NP problems have been proven to be *NP-complete*, i.e., they are maximally hard NP problems. A polynomial-time algorithm for an NP-complete problem would give polynomial-time solutions for all NP problems.

- **Open Problem:** Is factorization NP-complete?
George Dantzig invented the Simplex Algorithm shortly after the war which had interrupted his graduate studies at Berkeley. Usually it is very efficient, running in almost linear time. However for some problems, it takes exponential time.

A major theoretical problem was whether or not is was an algorithm which always solved linear programming problems in polynomial time. One was found, an ellipsoid method, which in 1979 by Khachian. It ran poorly on practical problems but later Karmarkar and others found algorithms which are efficient both theoretically and practically. However, the Simplex Algorithm is still the most popular and, on most problems, still the fastest.

The largest linear programming problems arise in the construction of economic models used to optimize allocation of resources in companies and countries. Linear programming models for oil refining have as many as 10,000 variables and equations.

On the other hand, integer programming has been proven to be NP-complete. Hence, assuming $\text{NP} \neq \text{P}$, there is no polynomial-time algorithm for solving integer problems. This remains true even if one restricts the integer variables to 0-1 decision variables. This wasn’t surprising since the problem of determining if a propositional formula is satisfiable is NP-complete.
From the Practice Exam

Classify as
knapsack,
stock cutting,
assignment,
shipping,
traveling salesman,
max flow,
shortest distance (directed or undirected),
longest distance (directed or undirected).
There are 9 grain storage areas: \(a, b, c, d, e, f, g, h, i\).

A “2” in the \(a-b\) box means you can directly ship one ton of grain from \(a\) to \(b\) for $20. For each storage area, find the minimum cost (in $10’s) to ship one ton of grain from \(a\) to that area.

Classify as knapsack, stock cutting, assignment, shipping, traveling salesman, max flow, shortest distance or longest distance (directed or undirected).
There are 9 grain storage areas: \(a, b, c, d, e, f, g, h, i\).

A “2” in the \(a-b\) box means 2 tons can be shipped from \(a\) to \(b\).

Find the maximum amount of grain which can be shipped from \(a\) to \(i\)?

Classify as knapsack, stock cutting, assignment, shipping, traveling salesman, max flow, shortest distance or longest distance (directed or undirected).
A bowl with 20 cups of soup is passed among some of the 9 people $a, b, c, d, e, f, g, i$. It starts with “$a$” and stops at “$i$”.

A “2” in the $a$-$b$ box means “$a$” takes 2 cups from the bowl and then passes it to $b$.

Find the minimum amount of soup which might be left in the bowl when it reaches $i$.

Classify as knapsack, stock cutting, assignment, shipping, traveling salesman, max flow, shortest distance or longest distance (directed or undirected).
Classify as knapsack, stock cutting, assignment, shipping, traveling salesman, max flow, shortest distance or longest distance (directed or undirected)

- Modular homes are made in plants $P, Q, R, S$ and then shipped to housing sites $A, B, C, D$. Each plant makes one house and each site gets one house. The shipping costs from a plant to a site are given in the matrix.
  
  Which plants should ship to which sites in order to minimize shipping costs?
  Find the minimum shipping cost.

- Nails are made in plants $P, Q, R, S$ and then shipped to housing sites $A, B, C, D$. The shipping costs from a plant to a site are given in the matrix. Also given are the amounts of nails (in bushels) produced by the plants and needed by the housing sites.
  
  What is the minimum shipping cost?
  How many nails should be shipped from each plant to each housing site?