Dynamic Programming:

1. Activity Selection

Instance: A list of activities ai, 1 ≤ i ≤ n with starting times si and ﬁnishing times fi. No two activities can take place simultaneously.

Task: Find a subset of compatible activities of maximal total duration.

1. Longest Increasing Subsequence

Given a sequence of n real numbers A[1..n], determine a subsequence (not necessarily contiguous) of maximum length in which the values in the subsequence are strictly increasing.

1. Making Change

You are given n types of coin denominations of values v(1) < v(2) < ... < v(n) (all integers). Assume v(1) = 1, so that you can always make change for any integer amount. Give an algorithm which makes change for any given integer amount C with as few coins as possible, assuming that you have an unlimited supply of coins of each denomination.

1. Integer Knapsack Problem (Duplicate Items Allowed)

You have n types of items; all items of kind i are identical and of weight wi and value vi. You also have a knapsack of capacity C. Choose a combination of available items which all ﬁt in the knapsack and whose value is as large as possible. You can take any number of items of each kind.

1. Integer Knapsack Problem (Duplicate Items NOT Allowed)

You have n items (some of which can be identical); item Ii is of weight wi and value vi. You also have a knapsack of capacity C. Choose a combination of available items which all ﬁt in the knapsack and whose value is as large as possible.

1. Balanced Partition(q3)

You have a set of n integers. Partition these integers into two subsets such that you minimise |S1 −S2|, where S1 and S2 denote the sums of the elements in each of the two subsets.

1. Assembly line scheduling

Two assembly lines with workstations for n jobs.

1. Matrix chain multiplication

Problem Instance: A sequence of matrices A1A2...An;

Task: Group them in such a way as to minimise the total number of multiplications needed to ﬁnd the product matrix.

1. Longest Common Subsequence

Task: Find a longest common subsequence of S,S∗.

1. Shortest Common Supersequence

Find a shortest common super-sequence S of s,s∗, i.e., the shortest possible sequence S such that both s and s∗ are subsequences of S.

1. Edit Distance

Given two text strings A of length n and B of length m, you want to transform A into B. You are allowed to insert a character, delete a character and to replace a character with another one. An insertion costs ci, a deletion costs cd and a replacement costs cr.

1. Maximizing an expression

Instance: a sequence of numbers with operations +,−,× in between, for example 1 + 2−3×6−1−2×3−5×7 + 2−8×9 Task: Place brackets in a way that the resulting expression has the largest possible value.

1. Turtle Tower

Instance: You are given n turtles, and for each turtle you are given its weight and its strength. The strength of a turtle is the maximal weight you can put on it without cracking its shell.

Task: Find the largest possible number of turtles which you can stack one on top of the other, without cracking any turtle.

1. Bellman Ford algorithm

A directed weighted graph G = (V,E) with weights which can be negative, but without cycles of negative total weight and a vertex s ∈ V . Goal: Find the shortest path from vertex s to every other vertex t.

1. Floyd Warshall algorithm

Let again G = (V,E) be a directed weighted graph where V = {v1,v2,...,vn} and where weights w(e(vp,vq)) of edges e(vp,vq) can be negative, but there are no negative weight cycles.

1. Relaxation

Compute the number of partitions of a positive integer n. That is to say the number of distinct sets of positive integers {n1,...,nk} which sum up to n, i.e., such that n1 + ... + nk = n.

