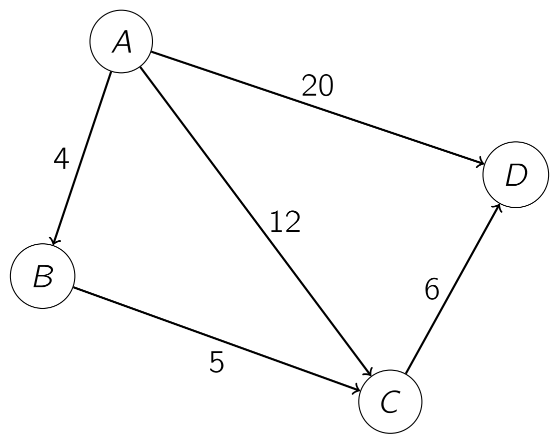
CS 161 (Stanford, Fall 2025) Section 6

1 Algorithm Practice

Given the directed graph below, run the Floyd-Warshall Algorithm, processing vertices in alphabetical order. Fill in the table below which keeps track of the shortest paths. Ordered pairs of vertices with no directed path (such as (B, A)) are omitted and their distance can be taken as ∞ for updates.



(u, v)	(A, A)	(A, B)	(A, C)	(A, D)	(B, B)	(B, C)	(B, D)	(C, C)	(C, D)	(D, D)
$D^{(0)}$	0	4	12	20	0	5	∞	0	6	0
$D^{(1)}$										
$D^{(2)}$										
$D^{(3)}$										
$D^{(4)}$										

2 Knapsack

Consider an instance of the knapsack problem with five items (where there is only one copy of each item):

Item	**Value**	**Size**
1	1	1
2	2	3
3	3	2
4	4	5
5	5	4

and knapsack capacity $C = 9$.

What are the final array entries of the Knapsack algorithm from lecture, and which items belong to the optimal solution?

3 Encoding

Suppose we encode lowercase letters into a numeric string as follows: we encode a as 1, b as 2, ..., and z as 26. Given a numeric string S of length n, develop an $O(n)$ algorithm to find how many letter strings this can correspond to. For example, for the numeric string 123, the algorithm should output 3 because the letter strings that map to this numeric string are abc, lc, and aw.

4 Knight Moves

Given an 8×8 chessboard and a knight that starts at position $a1$, devise an algorithm that returns how many ways the knight can end up at position xy after k moves. Knights move ± 1 squares in one direction and ± 2 squares in the other direction. In other words, knights move in a pattern similar to a "L".

Note: on a chessboard, rows are labeled from 1-8 and columns are labeled from $a - h$.