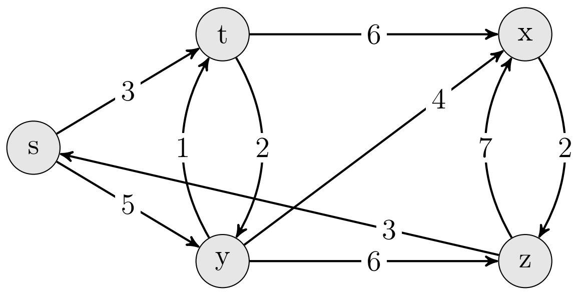
Exercise XI, Algorithms 2024-2025

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. There are many problems on this set, solve as many as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

Shortest Paths

1 (*, Exercise 24.1-6) Suppose that a weighted directed graph $G = (V, E)$ has a negative-weight cycle. Give an efficient algorithm to list the vertices of one such cycle. Argue that your algorithm is correct.

2 (Exercise 24.3-1) Run Dijkstra's algorithm on the following edge-weighted directed graph. Use first vertex s as a source and then vertex z as a source.



3 (half a *) Suppose you are standing at the top station of Mount-Everest and you wish to ski down. There are several different stations where you can rest (vertices) and different routes between stations (directed edges). All routes go down hill so they are directed from the station of higher altitude to the station of lower altitude. Although there are different routes from the top station to other stations, if one continues to ski down one will sooner or later hit the base camp. As the view is very nice, we wish to make the ski route as long as possible. In other words, design and analyze an efficient algorithm that calculates the *longest* route to ski down starting at the top station of Mount-Everest and ending in the base camp.

Randomness, Hiring problem, Indicator variables, Hash-tables

4 (Exercise 5.2-4) Use indicator random variables to solve the following problem, which is known as the **hat-check problem**. Each of n customers gives a hat to a hat-check person at a restaurant. The hat-check person gives the hats back to the customers in a random order. What is the expected number of customers who get back their own hat?

5 (Exercise 5.2-2) In the hiring problem, assuming that the candidates are presented in a random order, what is the probability that you hire exactly twice?

6 (Exercise 11.2-1) Suppose we use a hash function h to hash n distinct keys into an array T of length m. Assuming simple uniform hashing, what is the expected number of collisions? More precisely, what is the expected cardinality of $\{\{k, l\} : k \neq l \text{ and } h(k) = h(l)\}$?

7 (Exercise 11.2-2) Demonstrate what happens when we insert the keys $5, 28, 19, 15, 20, 33, 12, 17, 10$ into a hash table with collisions resolved by chaining. Let the table have 9 slots, and let the hash function be $h(k) = k \mod 9$.