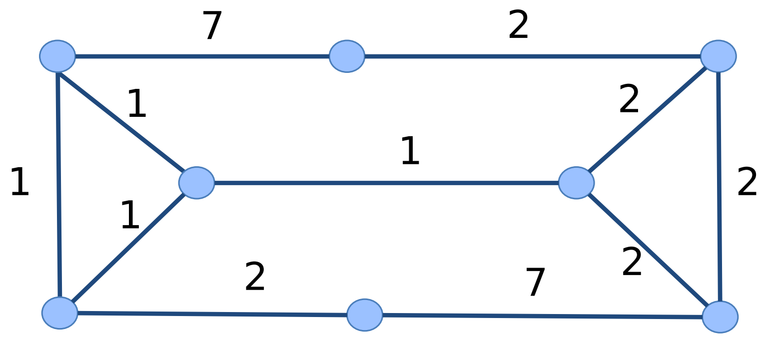
INTRO TO APPROXIMATION, CLASS 2

travelling salesman

Exercise one Consider the following graph:



1. List the shortest Hamiltonian cycle in this graph. (How would you verify that?)
2. What is the optimal solution to the graph TSP problem on this graph?
3. Choose a minimum spanning tree and construct the rest of the tour as Christofides' algorithm would.

Exercise two

Find a class of graphs showing that the algorithm for metric TSP that uses the doubling of the minimum spanning tree tour is no better than a 2-approximation.

In this exercise we are looking for an infinite class of graphs which has a strictly increasing number of vertices, i.e. we want $\{G_i | i \in \mathbb{N}\}$ so that $\forall i \in \mathbb{N}$: $|V(G_{i+1})| > |V(G_i)|$. That is a reasonable request; after all, if the tight bound would hold only for graphs with 20 vertices or less, and the algorithm would be 1.25-approximation for larger graphs, we would say that the algorithm is *asymptotically* a 1.25-approximation.

In this example we do not require tight bounds for small graphs; you should aim to prove that the doubling algorithm on a graph G_i is no better than a $(2 - x_i)$-approximation, where $x_i \to 0$. In other words, if your example will „clearly show“ that in the limit the bound is 2, you are done.

Exercise three Is TSP solvable in pseudopolynomial, or maybe fully polynomial time? One could suggest a dynamic programming algorithm as follows:

1. Create table $d[i, x, y]$ where the meaning of the entry is „best walk from x to y in i steps“.
2. Set $d[0, x, x] = 0$ and $d[0, x, y] = \infty$.
3. For every length $i \in \{1, 2, 3, \ldots, n\}$:
4. For every pair of vertices a, b:
5. Visit neighbors, set $d[i, a, b] = \min_{e | e \text{ edge connecting } a \text{ and } x}(d(e) + d[i-1, x, b])$.
6. Set $d[i, b, a] = d[i, a, b]$.
7. Return the minimum value $d[n, v, v]$ over all v.

Analyze this algorithm.

Exercise four Consider a connected, directed graph G. Before studying the TSP problem on directed graphs, we can search for a more general structure – a subgraph $P \subseteq G$ such that P contains all the vertices and every vertex has exactly one entering and one exiting edge.

This problem is called Minimum Directed Cycle Cover.

Hints:

- You can use a straightforward total unimodularity argument, if you know what that is from

Optimization methods.

- If you are not faimilar with total unimodularity, you can use a direct argument. You can for instance make use of the fact that minimum-weight perfect matching can be found in polynomial time. (Remember, Christofides' algorithm also uses this fact.)

EXERCISE FIVE ASYMMETRIC TSP is a problem tasked with finding the minimum TSP tour on a directed graph, where there is a different cost of travelling from a to b and back (but triangle inequality still holds).

1. First prove that finding the optimum of asymmetric TSP can be done using an algorithm finding the optimum of the symmetric TSP and vice versa – therefore, in a sense, the problems are equally difficult.
2. Next, prove that the two problems are not equally difficult at all – for instance show that the 2-approximation algorithm doubling the spanning tree does not make sense for the asymmetric case.