Review of Max Flow and All-Pairs Shortest Path Problems

Prof. Dr. M. Böhlen

April 14 2008

It is recommended that you solve the tasks without auxiliary material to mirror the setting at the exam. In general, when choosing among a number of nodes or edges use the (alphabetical) ordering as a tie-breaker.

Task 1:

Describe a solution for finding all pairs of shortest path in a directed graph. Provide a recursive implementation of your solution. Determine the complexity of your solution.

Task 2:

A number of students applied for summer jobs and received offers from companies. The following table shows the offers that students received. Describe how to determine an optimal assignment, i.e., an assignment that gives a summer job to as many students as possible.

	Adobe	Apple	HP	IBM	Sun	Yahoo
Alice	x	x	x			
Bob	x	x				x
Carol			x	x	x	
Dave	x	x				
Eliza				x	x	x
Frank			x		x	x

Task 3:

Assume a flow network $G = (V, E)$ with source $s \in V$ and sink $t \in V$. Prove that the Ford-Fulkerson algorithm never yields a flow with a value $f(u, s) > 0$.

Task 4:

Illustrate how you use the Ford-Fulkerson algorithm to find a maximal matching in the bipartite graph $G = (V, E)$ with $T = \{1, 2, 3, 4\}$ and $E = \{(1, 2), (2, 1), (1, 4), (4, 1), (2, 3), (3, 2)\}$. Show all steps of the computation.

Task 5:

Assume a flow network $G = (V, E)$. Let $f1$ and $f2$ be two flows in G. Consider the flow sum $(f1 + f2)(u, v) = f1(u, v) + f2(u, v)$. Is $f1 + f2$ a flow in G? Prove your answer.

Task 6:

Consider all pairs of shortest path in a directed graph. The predecessor matrix is defined as follows:

$$\pi^k[i,j] = \begin{cases} k & \text{if } d^{k-1}[i,k] + d^{k-1}[k,j] < d^{k-1}[i,j] \\ \pi^{k-1}[i,j] & \text{otherwise} \end{cases}$$

$\forall i, j : \pi^0[i, j] = 0$.

Assume π^6 looks as follows:

	1	2	3	4	5	6
1	0	5	0	5	0	0
2	4	0	0	0	4	0
3	6	6	0	6	6	0
4	0	5	0	0	1	0
5	4	0	0	2	0	0
6	4	0	0	2	4	0

Determine the shortest path from 1 to 6, from 2 to 5, and from 3 to 1. Write a procedure that takes a start and end vertex as arguments and prints out the shortest path between the two vertexes.

Task 7:

Give an algorithm that determines the length of a minimum-length negative-weight cycle in a directed weighted graph. Determine the asymptotic complexity of your algorithm.