# CS2223
# HW#7 SOLUTIONS

1. Reduce the instance of the PROGRAMMING_PACKING problem to an instance of (0/1)-KNAPSACK such that

| KNAPSACK | PROGRAM_PACKING |
|---|---|
| $n$ | $n$ |
| $W$ | $D$ |
| $v[i]$ | $s_i$ |
| $w[i]$ | $s_i$ |

The "weight" and "value" of each program is its size, and the capacity $D$ of the disk corresponds to the capacity $W$ of the knapsack. The optimal packing produced by the solution to the instance of knapsack problem yields an optimal packing of programs into the disk.

2. The conjecture is true. For any $v$, let edge $(v,w)$ be a minimal length edge (of length $l$) incident with $v$. Since all paths from $v$ to $w$ not including the edge $(v,w)$ start with a length greater than $l$, and all lengths are positive, the rest of the path makes it even longer, the length of a shortest path from $v$ to $w$ is $l$. *Floyd*'s algorithm returns $D[v,w]=l$.

3. The conjecture is false. The triangle (complete graph on three vertices) with edge lengths 5, 6 and 40 is a counterexample.

4. Do a depth first search to check if the graph is connected, but quit as soon as a cycle is found. If the graph is connected and does not have a cycle, then it's a tree, otherwise it's not a tree. Since a cycle, if it exists, will be found after examining at most $n$ edges, the algorithm works in time in $O(n)$.