1 Hints to Assignment 1

The following hints are useful in solving some of the problems assigned in the homework.

1.2 The main observation is that the rows of the preference lists for men and women can be permuted. See what happens when m and w are the first rows of their respective lists, and the algorithm of G-S is applied.

1.3 Show that there is not always a stable pair of schedules. This is due to the arbitrary nature of the ratings for the show. You should be able to come up with an example where each network has two shows.

1.6 For each ship a stopping port has to be chosen where it will spend the rest of the month. The stopping ports define truncations of the schedules. Each ship ranks each port in chronological order of its visit. Each port also ranks each ship in chronological order of their visits. Now show that the G-S algorithm provides a stable matching.

1.7 It is very similar to the problem 1.6.

Problems of Chapter 2 These problems are easy to solve.

Problem 3 You need to show that there are $\Theta(n^2)$ unmatched pairs to be tested. Note that $.0001 * n^2$ is still $\Theta(n^2)$.

2 Hints to Practice Problem

There are 6 unstable pairs.

You can always use G-S male-optimal and woman-optimal algorithm. If they are the same, then you need to argue that the stable matching is unique.

3 On topics covered in the last lecture

1. We have considered the following problem in the class.

   A sink in a simple (no self-loops) digraph $G = (V, E)$ is a vertex with $|V| - 1$ incoming edges and no outgoing edges. Find an algorithm that, given the adjacency matrix A of G, determines in $O(|V|)$ time whether G contains a sink.

(Solution) The entries of the adjacent matrix are as follows. If there is a directed edge from node i to node j, $a_{i,j} = 1$, otherwise $a_{i,j} = 0$. We showed that if we consider an element $a_{i,j}$ element of A, then node i cannot be a sink if $a_{i,j} = 1$, and node j cannot be a sink if $a_{i,j} = 0$. Thus after looking at appropriate $|V| - 1$ elements of A, we are able to eliminate $|V| - 1$ nodes from being a sink. The remaining node is tested for the sink by visiting its column in the adjacency matrix.

2. Consider an arbitrary graph $G = (V, E)$. Let $n = |V|$ and $m = |E|$. The lower bound value of m is zero (the graph has no edge). The upper bound value of m is $\Theta(n^2)$ (the graph is dense). The adjacency matrix of G requires $\Theta(n^2)$ space, where as the adjacency list requires $O(m + n)$ space. The list representation is storage space optimal. We should decide on the representation depending on the problem we are working on.

3. We have seen how a heap can be stored in an array. The left child (right child) of the i^{th} node is located at location $2 * i$ $(2 * i + 1)$, if $2 * i \leq n$ $(2 * i + 1 \leq n)$. The heap structure is a full balanced binary tree. A balanced tree has a $3 - ary$ heap structure if it is balanced, and all the non-leaf nodes (except possibly one at level $h - 1$ where h is the height of the tree) have 3 children, and all the leaf nodes are at adjacent levels. We can store this heap structure in an array A where $A[1]$ contains the root node, the 3 childrent of the root node are at locations $A[2], A[3], A[4]$, and so on. If we store the nodes this way, show that 3 children of the object at location i will be placed at locations $A[3*i-1], A[3*i], A[3*i+1]$. I am assuming that $3*i+1 \leq n$. Also show that the parent of $A[i]$ is at location $\lfloor (i + 1)/3 \rfloor$. Can you generalize the mapping for any $k - ary$ heap structure?