Homework 5

Try to be precise and to the point. Your answers should be short.

1. Run the $Dijkstra(G, s)$–algorithm on graph G in Figure 1 of Lecture Note 9, starting at v. You need to draw a table (similar to the table in LN9.pdf).

   Solution: Easy.

2. Let T be a heap tree (representing an array H) of height n.
   (a) What is the number of internal nodes of T with exactly one child? Explain your answer.
   (b) Explain why every leaf of T is at distance n or $n-1$ from the root of T.

   Solution: (a) At most 1. Indeed, all the items with index i in heap array have parents. If i is odd then $i-1$ is the index of its sibling. Hence, the only node that has no sibling is the last item in the heap with even index.

   (b) Let T be a tree representing the array H. Let k be the number of elements in H. Nodes at positions 2,3 are at level 1 of the tree, nodes at positions, $4, 5, 6, 7$ are at level 2 of the tree, etc. So nodes at positions $2^i, 2^i + 1, \ldots, 2^{i+1} - 1$ are at level i of the tree. So, if n is the height of T then there are exactly 2^{n-1} elements at level $n-1$ of the tree; these elements are at positions $2^{n-1}, 2^{n-1} + 1, \ldots, 2^n - 1$ of the tree. Since n is the height of the tree, the array H has elements at positions $2^n, 2^n + 1, \ldots, k$, and $k < 2^{n+1}$. This implies that every leaf of T is either at level n or level $n-1$.

3. Let G be a graph. A k-colouring of G is a colouring of vertices with at most k colours such that no two adjacent vertices have the same colour.
   (a) Describe 1-colourable graphs.
   (b) What is the minimal number of colors needed to colour a tree with more than 1 node. Explain your answer.
   (c) Explain that the bipartite graph $K_{n,m}$ can be 2-colourable.

   Solution: (a) all graphs that have no edges. (b). Color the root white. Proceed as follows. If v is colored black then color its children white. If v is colored white then color its children black. So two colors suffice to color a tree. (c) $K_{n,m}$ is a complete bipartite graph with n+m vertices. Colour all vertices in one side of the graph with white, and color all vertices in the other side with black.

4. Let k be the maximal among the degrees of all vertices of graph G. Write down a greedy and linear time time algorithm that colours G with $k+1$ colours.

   Solution: List all vertices of the G: v_0, v_1, …, v_i, …, v_n. Color v_0 with 1. For v_i color it with the first available color. Since the degree of each vertex is not more than $k+1$, it is clear that the process described colour the graph with at most k colors.

5. Suppose $\mathcal{S} = \{a, b, c, d, e, f\}$. Give two examples of prefix codes for these letters. Present the prefix codes as binary trees.

   Solution: Easy. Just draw two binary trees with exactly 5 leaves. each path in this tree would code a letter.