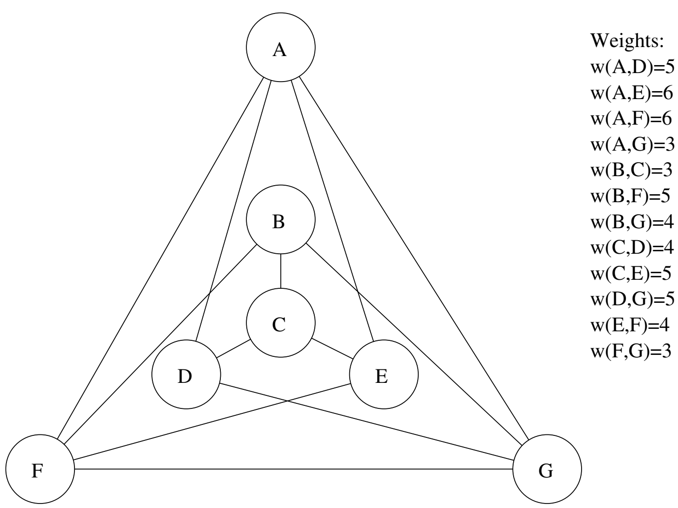
Cs445 — Homework #3
Radix Sort, MST and Huffman codes

Due: 28 February 2007 during class meeting.
(Revised 2/21)

Instructions. All assignments are to be completed on separate paper. Use only one side of the paper. Assignments will be due at the beginning of class. To receive full credit, you must show all of your work. All questions have the same value.

1. Run Prim's algorithm on the following graph, shown in the image. Show every stage of the algorithm, and the values of the array $key[]$ after each stage.



2. You are given a set $\{w_1 \ldots w_n\}$ of strings. Each string w_i is over the alphabet $\Sigma = \{a \ldots z\}$. Let $m_i = |w_i|$ denote the length of w_i, and let $m = \sum_{i=1}^{n} m_i$. Suggest an algorithm, based on a simple modification of the Radix sort algorithm, that lexicographically sorts all the strings, and runs in time $O(m)$. Note that an algorithm that runs in $\Theta(n \cdot \max\{m_1, \ldots m_n\})$ will not be accepted.

   Example for the requirement of sorting lexicographically: The word "aaa" is smaller than "aab" which is smaller than "aaba".

3. Let n be a given integer, which is not specified. Suggest an example of a graph $G(V, E)$ with $|V| = n$ nodes and $|E| = \binom{n}{2}$ edges, for which executing Prim's algorithm would actually execute $\Theta(n^2)$ changes of the $key[]$ values.

   Hint: Assume V consists of two disjoint subsets $V = V_1 \cup V_2$, each containing $n/2$ vertices. Create the weighs of the edges so that the vertices of V_2 are added to the MST only after the vertices of V_1, but every time that the algorithm adds a vertex of V_1 to the MST, it causes a change to all $key[v]$ values for all $v \in V_2$.

4. The question is motivated by the task of maintaining MST in a network where edges appear and disappear. Let $G(V, E)$ be a graph, with weights assigned to its edges, and let T be its MST, which is given to you. Assume that $e = (u, v)$ is an edge that is **added** to the graph after T is constructed. Suggest an algorithm that finds in time $O(n)$ a MST of the graph $G(V, E \cup \{(u, v)\})$, where $n = |V|$.

5. Discuss all the needed data structures that are actually needed in order to run Prim's algorithm (the ones presented in the slides skip several of them).

6. Run Huffman's algorithm on the following phrase: *In the Tian Shan, I stash tea in a thin tin.* Ignore spaces and punctuation, i.e., encode only the following:

   `INTHETIANSHANISTASHTEAINATHINTIN`

7. In the slides the about Huffman code algorithms, the function $f(x)$ (for any character x in the alphabet) is defined as the number of appearances of x in the input file to be encoded. A different definition that is commonly used for $f(x)$ is the **frequency** of x in the file, that is, the ratio of the number of times x appears in the file to the number of characters in the file.

   Prove the these two definitions are equivalent; that is, a code is optimal when using the first definition if and only if it is optimal when using the other definition.

8. You have a sufficient supply of coins, and you goal is to be able to pay any amount between 1 cent and 99 cents using as few coins as possible. That is, the input to the algorithm is an amount, x cents, $x \leq 99$, and the output is the number of coins of each type so that the total value is x cents, and the total number of coins is as small as possible.

   (a) Describe a greedy algorithm to solve the problem when the coins used are of values $1, 5, 10$ and 25 cents. Prove the optimality of the algorithm.

   (b) Show that a greedy algorithm does not yield optimal results if you are using only coins of value 1 cent, 7 cents and 10 cents.

9. **Bonus** You are given a building with n stories and no elevator, and two identical bottles. You want to determine the maximum height h^* from which you can drop a bottle to the ground without it breaking. Let m be the total number of stories you have climbed during the experiment. Describe a strategy to find h^* while climbing a total of $O(n\sqrt{n})$ stories.

   Note: Once a bottle is broken, you cannot use it again. If you want to use a bottle that you dropped and that did not break, you have to fetch it, and climb up again. You start your experiments when you are on the ground floor.