CSE 326: Data Structures
Assignment #7
December 4, 1998

You do not have to turn in this assignment. Its purpose is to give you practice for the final exam on recent topics.

1. Consider the universal class consisting of the hash functions

$$h_{a,b}(x) = ((ax + b) \bmod 71) \bmod 11$$

for $0 < a < 71$ and $0 \leq b < 71$. Let x_1 and x_2 be any two distinct keys in $\{0, 1, \ldots, 70\}$. Justify your answers to the following. (Hint: you definitely don't want to start listing all the (a, b) pairs, as there are 4970 of them. Instead, use what you know from the proof of the Universal Classes of Hash Functions Theorem.)

   (a) Exactly how many of the $70 \cdot 71$ (a, b) pairs hash both x_1 and x_2 into bucket 4 (where the buckets are numbered $0, 1, \ldots, 10$ according to the value of $h_{a,b}(x)$)? Exactly how many hash both x_1 and x_2 into bucket 5?

   (b) Exactly how many ordered pairs of distinct numbers (q, r) are there such that $0 \leq q, r < 71$ and $q \equiv r \pmod{11}$? Compare your answer to the upper bound $N(N-1)/m$ proved in the Theorem.

   (c) What is the exact probability that a randomly chosen hash function $h_{a,b}$ will cause x_1 and x_2 to collide? Compare your answer to the upper bound $1/m$ proved in the Theorem.

2. Page 331, problem 1. Show the partially ordered tree after each Insert and after each DeleteMin. Also show the heap in table form after the last Insert.

3. Page 331, problem 2.

4. (a) Simulate Dijkstra's algorithm on the following graph to find the least cost path (the path itself, not just its cost) from vertex A to every other vertex. Each time a vertex is removed from U, do the following: redraw the vertices (but not the edges) of the graph, circle the vertices that still remain in U, show the current values of Distance(v) next to each vertex v, and draw an edge from v to the current value of Pred(v) for each vertex v for which $0 <$ Distance$(v) < \infty$. (As in lecture, Pred(v) is the immediate predecessor of v on the least cost path from A to v found by the algorithm so far.) For each of these steps, also show the

partially ordered tree corresponding to U, labeling each node with both its Key and Info values.

(b) From your output in part (a), how would you find the least cost path from A to D?

5. (a) Simulate Kruskal's algorithm to find a minimum cost spanning tree T of the following graph. Each time an edge e is added to T, redraw the current forest T and the full collection of up-trees resulting from adding e to T. Each time an edge e is considered but not added to T, list the Find operations that explain why e was not included in T. Use Weighted Unions, but without path compression.

(b) What is the cost of the minimum spanning tree that you found?