Midterm #2
CIS 121—Fall 2016

In-class exam: Thursday, November 3rd, 2016.
Exam Starts: 10:30AM
Exam Ends: 11:50AM

This is a closed book exam. No computers or internet-connected devices are allowed during the exam. You are permitted to use one 8.5”x11” page with handwritten notes (these can be on both sides of the paper). If you need scratch paper, please get some from the front of the classroom. There are 8 questions on this exam, plus 2 extra credit questions, for a total of 10 questions. May the Force be with you.

Your name:

Your PennKey (this should be your username like ccb, not a number):

(circle one) Mon 12-1 Mon 2-3 Mon 4-5 Mon 5-6 Tues 12-1 Tues 2-3 Tues 3-4

Tues 4-5 Tues 5-6 Mon 1-2 Mon 3-4 Mon 6-7 Tues 6-7 Mon 11-12 Tues 1-2
1 Graphs
10 points

For each of the statements below, please say whether it is true or false, and give a 1 sentence explanation of your answer.

1. For a connected undirected graph with \( N \) vertices, the minimum number of edges that it can have is \( N - 1 \).

2. For a connected undirected graph with \( N \) vertices, the maximum number of edges that it can have is \( N^2 \). Assume that we allow self-loops but not parallel edges.

3. In a typical adjacency matrix representation of a graph with \( N \) vertices and \( M \) edges, the amount of space required is \( N^2 \).

4. If all edges in a graph have distinct weights, then the shortest path between two vertices is unique.

5. A boolean adjacency matrix representation of a graph supports parallel edges.
2  Dijkstra’s running time
10 points

• Give a short English description of Dijkstra’s shortest path’s algorithm.

• What is the worst case running time of Dijkstra’s shortest path algorithm for a connected graph with $E$ edges and $V$ vertices when using a binary heap for the priority queue? What are the operations? How many times is each operation called? What is the running time for each operation?
3  **TSTs**  
10 points

1. What are the results of searching for the following keys?
   (a) CAT
   (b) ACT
   (c) CAG

2. Insert the key CCB with value 11 into the TST above. (You can draw on the figure itself. Be sure to draw all children, including null links).

3. What is the shortest number of steps needed for a search hit for a key of length \( L \) in a TST? When is it guaranteed to occur?
4 R-way tries

10 points

1. Please define R for an R-way trie.

2. What is the worst case running time for inserting a value for a key of length $L$? Please give your answer in big O notation, explaining any variables that you use, and saying when the worse case happens for insertion.

3. What is the best case running time for inserting a value for a key of length $L$?

4. What operations can take less than $L$ time, and when do these occur?
5 Balanced trees
15 points

1. Which of the following trees are balanced? Write “balanced” or “not balanced” next to each one. Give a short explanation about how it is balanced, or give a valid counterexample if it is not balanced.

(a) Binary search trees
(b) Heaps
(c) Red-black trees
(d) 2-3 trees
(e) R-way tries
(f) Ternary search tries
(g) Subtrees corresponding to the connected components in the weighted Quick-Union algorithm

2. What is the correspondence between the height of a balanced tree with $C$ children, the number of elements stored in it, and the log function? As a reminder, $\log_b(x)$ is the unique real number $y$ such that $b^y = x$. For example, as $64 = 2^6$, then: $\log_2(64) = 6$.

3. What impact does the height of a tree have on the worst case running time for standard search tree operations?

4. When does the height of a search tree change? Give your answer for a 2-3 tree and for a red-black tree.
6 Linear Probing Hash Table  
15 points

<table>
<thead>
<tr>
<th>keys[]</th>
<th>P</th>
<th>M</th>
<th>A</th>
<th>C</th>
<th>S</th>
<th>H</th>
<th>E</th>
<th>R</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Given the linear probing hash table above, record where the following keys would be added. Write the key into the appropriate slot in the table, assume that they are added in the order that we give them in.
   (a) Insert Z. Hash(Z) = 9.
   (b) Insert Y. Hash(Y) = 6.
   (c) Insert C. Hash(C) = 4.
   (d) Insert K. Hash(K) = 14.

2. Hash tables store values associated with keys. How are values stored in a linear probing hash table?

3. What is the maximum number of keys that the linear probing hash table above could store? Why would it be undesirable to store that many keys in the hash table?

4. Name another method that can be used to resolve collisions that result from the hash function, and give a pro and a con for using it over linear probing.
7 Simple Data Structures Support Complex Algorithms

15 points

Kruskal’s algorithm for computing a minimum spanning tree can be described as follows:

- Consider edges in ascending order of weight
- Add the next edge to the tree unless doing so would create a cycle.
- Stop when we have added enough edges to form a spanning tree.

Say what data structures an implementation could use to support the below operations, and what their running time would be for those operations. If multiple data structures could be used, then pick the one with the smallest running time.

1. Ordering all of the edges in an undirected graph, starting with the edge with the lowest weight and going in order to finally return the edge with the highest weight.

2. Finding whether there is a cycle formed when adding an edge to the partial MST that we have constructed so far, and adding that edge to the data structure if it doesn’t create a cycle.

3. Returning the set of edges that form the MST in the order that Kruskal’s algorithm added them to the MST.
8 Make money with an app
15 points

An entrepreneurial team of Penn undergrads builds an app using Dijkstra’s shortest paths algorithm (not the buggy version in the slides). Their app lets drivers find the shortest distance to their destination.

1. How is the system of roads represented in the graph? What are the vertices and edges? What type of graph is it?

The US congress has decided that in order to reduce the national deficit, all roads are now toll roads. Users of the app demand an update that will let them find the lowest cost route to their destination in terms of dollars instead of miles.

2. What do the Penn students need to do to update their graph?

The newly elected president decides to name the best, classiest road in each American city “Presidential Avenue”. The president decrees that instead of charging drivers a toll, the government will pay drivers whenever they drive on any street named Presidential Avenue.

3. Assuming that the graph automatically updates each time the tolls change, what happens to the edge weights in the graph? Can the app still give drivers the lowest cost ($) route to their destination?

4. How could the Penn students determine if the app could be fixed, or if it is impossible to correctly calculate shortest paths? If it is fixable, what is the solution?

5. If the app can no longer provide driving directions, how could the Penn students still use it to make money for themselves?
9 Extra Credit: Heaps with twice as many elements
5 points

Fifi wants to build a heap from two unsorted arrays, each of length $N$. She wants you to describe an algorithm that will combine the two arrays and put them into heap order. Describe the algorithm that you should use and its running time. Your answer should include brief descriptions of what heap order is, what sink and swim operators are, and what the running time is of applying sink or swim once.
10 Extra Credit: Picking the right tool for the job (2 points each)
12 points

Below there are a short description of programs that we would like to write. Please say which of the data structures or algorithms we learned so far in class would be most practical to use in each program. Give a 1 sentence description why you picked that one.

1. Address book autocomplete: based on typing the first few letters of a name, list all contacts stored in an address book that match those first few letters.

2. k-best list: Given a stream of a million numbers, keep track of the k highest numbers in a space efficient way.

3. Given Facebook’s graph (where friendship is an edge and each node is a person), find all the connected components. Pick an algorithm that allows new friendships to be formed over time. You don’t have to support ending friendships.

4. Spell check: given a dictionary containing correctly spelled words, check all words in a document, highlighting ones that do not match the dictionary.

5. Given a map of cities connected by one-way roads, we want to identify groups of cities such that for any pair of cities in the group, it is possible to drive from the first city to the second and vice versa.

6. We want to build a program by compiling individual files in some order. It may be the case that a file can only be compiled after certain other files have been compiled. How do we write a script that determines the correct order to compile the files? You can assume that there exists no cycles (e.g. it will not be the case that file A requires file B, but file B also requires file A).