CSE 674 Final Exam-ALT
Spring 2015

Name: _____

| Question | Weight | Score |
|---|---|---|
| 1 | 10 | |
| 2 | 10 | |
| 3 | 10 | |
| 4 | 10 | |
| 5 | 10 | |
| 6 | 10 | |
| 7 | 10 | |
| 8 | 10 | |
| 9 | 10 | |
| 10 | 10 | |
| Total | 100 | |

Question 1

Part A (1 point each) Circle the right answer. No explanation is required

1. We can use depth first search to detect if a given undirected graph is in fact a tree.
   True/False

2. The Hancart string matching algorithm performs better on average than the Knuth Morris Pratt algorithm
   True/False

3. Let G be an un-directed graph and H be a directed graph resulted from assigning directions to the edges of G. Then, H needs more space to store than G.
   True/False

4. A Bloom filter use a collection of hash functions to provide a definite answer to tell whether an element is in a given set
   True/False

5. The collection of strings, which are formed by concatenation a finite number of strings selected from 00,11 or 011, can be presented by a regular expression
   True/False

6. Let S be a skip list which stores n elements. To search for an element in S takes O(lg n) time.
   True/False

Part B (4 point) Draw a deterministic finite automata (DFA) M that can be used to identify any binary strings that has 1100 as a substring. Label all the features of your DFA M.

Question 2

The Bellman-Ford Algorithm and the Matrix Chain Multiplication problems are two examples we discussed regarding the method of dynamic programming.

1. (2 point) Under what circumstances will you use Bellman-Ford algorithm, instead of Dijkstra's algorithm, to solve the single source shortest path problem?

2. (2 point) Let G=(V,E) be the underlying graph. State the running time for Bellman Ford algorithm in terms of |V| and |E|.

3. (4 point) Let A, B and C be three matrices where the product ABC is well defined.
    a. Give an example to show that the number of scalar multiplications performed by (AB)C is larger than the number of scalar multiplications performed by A(BC).
    b. Given another example to show that the number of scalar multiplications performed by (AB)C is smaller than the number of scalar multiplications performed by A(BC).

4. (2 point) Why won't we use a divide and conquer approach to find a solution for the matrix chain multiplication problem? Give a concise explanation.

Question 3 (10 point)
Given 1MB (1024K) memory (initially unused), a system applies the buddy system method to allocate and de-allocate memory to meet the following requests:

A: request 100k; B: request 240k; C: request 64; D: request 256k; release B; release A;
E: request 75k; release C' release E; release D

Illustrate, with the aid of diagram(s), to show how the buddy system works for the above example. Include all the intermediate steps in your illustrations.

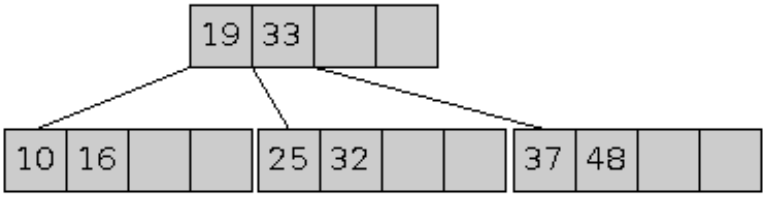
Question 4 (10 point)

Given the following frequency table for the characters a to g:

| Letters | a | b | c | d | e | f | g |
|---|---|---|---|---|---|---|---|
| Frequency | 20 | 14 | 12 | 16 | 25 | 8 | 5 |

Apply the Huffman coding method to construct a Huffman code for the given set of characters. State the name of a data structure that you use in the implementation of the method (no need to show it in your illustrations).

Question 5 (10 point) Present the resulting B-Tree after the key 33 is deleted from the B-tree below. Show all the intermediate steps (including the key copy).



Question 6 (10 point) Insert the following sequence of numbers 16, 5, 7, 6, 19, 11, 13, 17, 23, 8 via double hashing into a hash table of size 11. The hash functions are:

hash1(x) = x % 11
hash2(x) = (x % 5) + 1

Show steps and calculations.

Question 7 (10 point) Apply counting sort (stable version) to sort the array below in ascending order. Include all the steps such as allocating additional memory. You may assume that the numbers in any input arrays is 1, 2, 3, 4 or 5.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 4 | 5 | 1 | 1 | 1 | 4 | 2 | 5 |

Question 8. (10 point)

Write code to sort a double linked list in ascending order (with respect to the key values). The sorting algorithm you use must be both *in-place* and *stable*. You must use the API given for full credit. Do not use any STL containers or STL algorithms.

```
1  class ListNode {
2     int key;
3     ListNode * prev;
4     ListNode * next;
5
6  };
7
8  void sort(ListNode * head){
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
```

```
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93  }
```

Question 9. (10 point)

Given a binary tree, write code to remove the node with a given key, if any. You may assume each node has a distinct key. Do not use the standard template library.

```
1  struct TreeNode {
2     int key;
3     TreeNode * left;
4     TreeNode * right;
5  };
6
7  void remove (TreeNode * theTree)
8  {
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
```

```
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92  }
```

Question 10 (10 point) Write code to find the smallest element from a max heap. Your code cannot search the entire heap. Use the API provided. You may assume that the heap is already a max heap and it is not null.

```
01  int findSmallest(std::vector<int> heap){
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
```

```
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
74
75
76
77
78
79
80
81
82
83  }
```