COMP 3270 Assignment 5 20 Multiple Choice Questions No late submissions!
Due before 11:59 PM Wednesday April 22

Instructions:

1. This is an individual assignment. **You should do your own work.**
2. Late submissions will **not** be accepted.
3. Mark answers in the this PDF file by using "Highlight Text" feature of Adobe Acrobar Reader.
4. Upload PDF file to Canvas with correct answers highlithed **before the due date & time**.
5. All questions carry 2 points.

Binary Search Trees (BST)

1. Draw the Binary Search Tree resulting from inserting 3,1,4,6,9,2,5,7 one after the other into an initially empty BST and then answer this question: **True or False?** The final BST will have a subtree rooted at 6 with a left child 5 and a right child 9, with 9 having a left child 7.

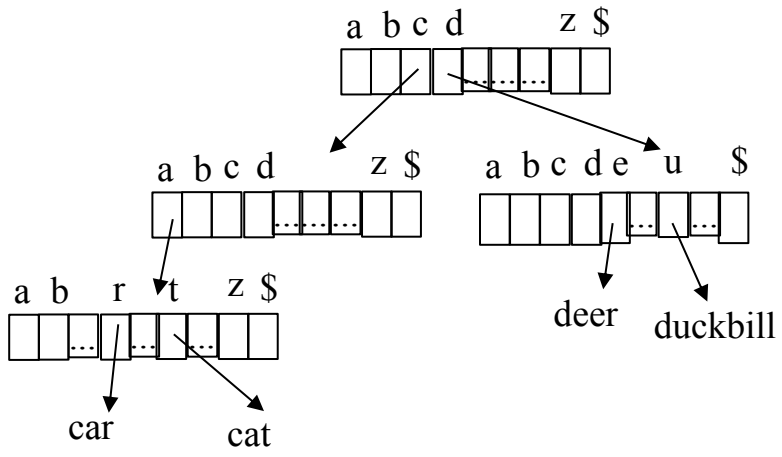
2. We can sort n numbers by first building a BST containing the n numbers (start with an empty BST and call Tree-Insert repeatedly n times) and then outputting the numbers from the BST using an Inorder-Tree-Walk. What is the complexity of this sorting algorithm?

A. O(n) B. O(n^2) C. O(logn) D. O(nlogn) E. none of these

3. Simulate the Tree-Delete algorithm (slide # 24) on slide set 16 Searching Part I) on the BST on slide 23 of slide set 16 with z=5. Which of the statements below is true?

A. Condition check on step 9 of the algorithm is the one that succeeds.
B. Condition check on step 11 of the algorithm is the one that succeeds.
C. Neither will succeed and step 13 will be executed.
D. Execution will never reach the if statement in steps 9-13.
E. None of the above statements is true.

4. Now simulate the TREE-DELETE algorithm (p.298 of the text) on the same BST on slide 23 slide set 16 with z=5 and answer this question: **True or False?** The tree resulting after deletion in this case is different from the tree resulting after deletion using the Tree-Delete algorithm of Q. 3 above.



5. Suppose the word "catcall" is to be inserted into the above Trie. How many additional internal nodes (i.e., nodes with an array of 27 pointers) will get added?

A. 1 B. 2 C. 3 D. 4 E. 5

String Searching

```
NAÏVE-STRING-MATCHER (T,P)
1 n = T.length
2 m = P.length
3 s = 0
4 while s<n–m+1 do
5       for i = 1 to m
6               if P[i] != T[s+i] then
7                       s = s+1
8                       exit the i-loop and go to step 4
9       print "pattern occurs with shift" s
10      s = s+1
```

6. If P=0001 and T=000010001010001, what are the shift values printed by step 9 of the algorithm above?

A. 0, 4 & 10 B. 2, 6 & 12 C. 5, 9 & 11 D. 1, 5 & 11 E. nothing will be printed

```
MODIFIED-NAÏVE-STRING-MATCHER (T,P)
1 n = T.length
2 m = P.length
3 s = 0
4 while s<n–m+1 do
5       for i = 1 to m
6               if P[i] != T[s+i] then
7                       s = s+i
8                       exit the i-loop and go to step 4
9       print "pattern occurs with shift" s
10      s = s+m
```

7. True or False? MODIFIED-NAÏVE-STRING-MATCHER is a correct algorithm.

Do problem 32.2-1 in the text p. 994 before answering questions 8 and 9.

8. How many spurious hits will there be?

A. 1 B. 2 C. 3 D. 4 E. no spurious hits

9. How many correct matches will there be?

A. 1 B. 2 C. 3 D. 4 E. no correct matches

10. Working modulo q=13, how many spurious hits does the Rabin-Karp algorithm encounter in the text T=16152279 when looking for the pattern P=27?

A. 1 B. 2 C. 3 D. 4 E. 0

Disjoint Sets

11. If the array below implements disjoint sets as trees, what is the union being implemented? The top row are cell indexes.

1	2	3	4	5	6	7	8
-2	6	8	8	1	3	1	-4

A. arbitrary B. by size C. by height

Design a recursive, and a non-recursive algorithm that employs a stack, for the operation **Find(e)** where e is a data element and Find is to be done with path compression. Assume the disjoint set is implemented using array P and union by size. Partial algorithms are given below.

Find-recursive(e, P)

1 if P[e] > 0 then
2 _____**Q 12**_____
3 else return e

12. What is (are) the correct step (steps) to fill the blank above?

A. Find-recursive(e, P)
B. P[e] = Find-recursive(e, P)
C. return P[e]
D. P[e] = Find-recursive(e, P); return P[e]
E. none of these

Find-iterative(e, P)

S: stack
1 while e is not negative
2 push(e, S)
3 _____**Q 13**_____
4 v = pop(S)
5 while S is not empty
6 _____**Q 14**_____
7 _____**Q 15**_____
8 return v

13. Which step should go to line 3 of Find-iterative?

A. P[temp] = v B. temp = pop(S) C. e = P[e] D. push(e, S) E. none of these

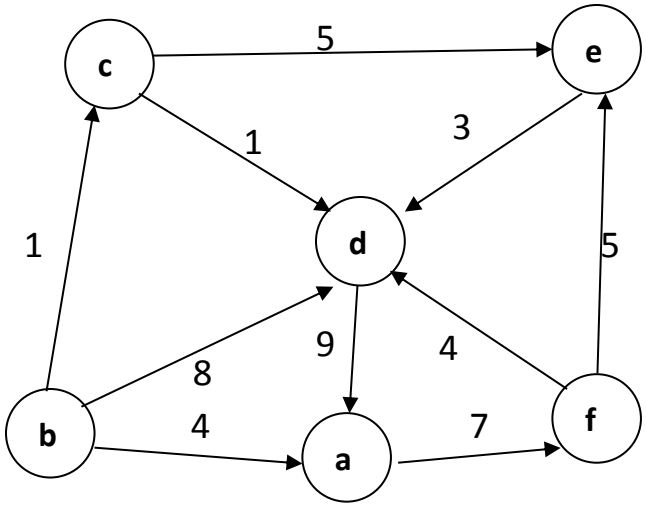
14. Which step should go to line 6 of Find-iterative?

A. P[temp] = v B. temp = pop(S) C. e = P[e] D. push(e, S) E. none of these

15. Which step should go to line 7 of Find-iterative?

A. P[temp] = v B. temp = pop(S) C. e = P[e] D. push(e, S) E. none of these

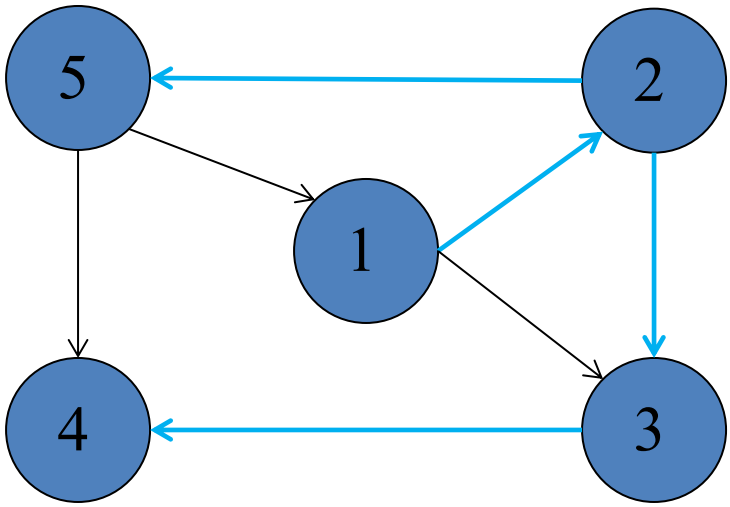
Graph representations and algorithms



16. Simulate algorithm Breadth-First Search (slide 7, slide set 20) on the graph above with "a" as the starting node and determine the d and ∏ values of each node. Assume that adjacent nodes are considered and inserted in the Queue in alphabetical order. What are the d and ∏ values of node "e"?

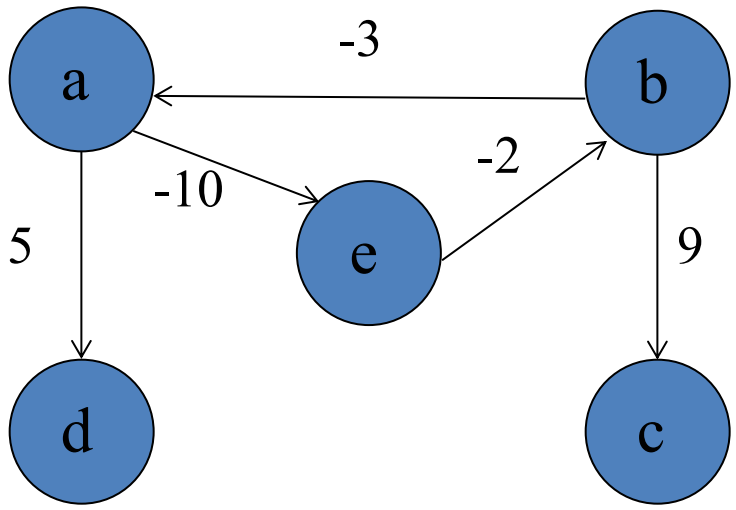
A. 2 & f B. 1 & a C. 0 & NIL D. ∞ & NIL E. none of these

17. If DFS-VISIT started at node 1 and the edges it traversed through recursive calls are shown in blue on the graph to the right, what are the correct classifications of the remaining three edges 5->1, 1->3, and 5->4 respectively?



A. Cross, Forward and Back.
B. Forward, Back, and Cross.
C. Cross, Back and Forward.
D. Back, Forward and Cross.
E. none of the above

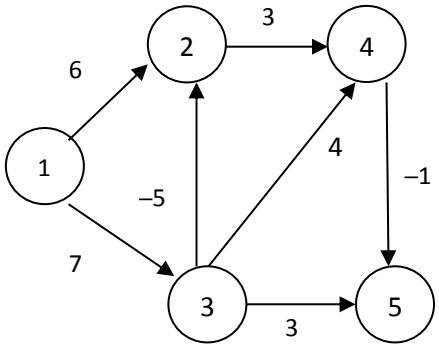
18. If the Bellman-Ford algorithm is run on the graph below with a negative cost cycle, with "a" as the start node, which edge when considered in step 6 of the algorithm will make the algorithm return false? Assume the for loop in step 5 considers the edges in the following order: a->d, a->e, b->c, b->a, e->b.



A. edge a->d edge B. a->e C. edge b->c D. edge b->a E. edge e->b

19. Do Problem 24.2-1 on p. 657 of the text. Then answer the following question. The for loop in step 3 of the DAG-SHORTEST-PATHS algorithm on text p. 655 will run six times. **True or False?** The "d" values of none of the six nodes will change during the fourth, fifth and sixth iterations of the for loop.

20. Consider the graph below with nodes numbered 1-5 (node numbers are inside the circles) with edge costs as shown. Simulate Dijkstra's algorithm on this graph with 1 as the start node, and answer the following question. The algorithm will fail to find the shortest paths because the graph has negative weight edges. That is, the final "d" values of nodes after the algorithm completes may or may not have the correct costs of the shortest paths to those nodes from the start node 1. The d-values of which nodes are incorrectly computed by the algorithm?



A. nodes 2 & 3 B. nodes 2 & 4 C. nodes 3 & 5 D. nodes 3 & 4 E. nodes 4 & 5