Travis' Exam Review

Monday, April 23, 2018 5:02 PM

Final Exam Review

1. (10 points) Prove or Disprove: $an^2+bn \in \Omega(n^2)$ for non-negative a and b.

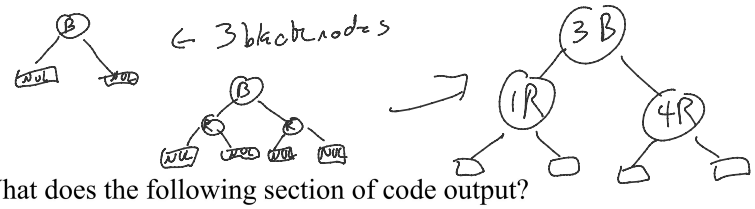
$\lim_{n\to\infty} \frac{an^2+bn}{n^2} = C > 0$ if $a = 0$

$a + \frac{b}{n} = a$ then $an^2+bn \notin \Omega(n^2)$

$a = 0$

2. (10 points) Draw a valid Red-Black Tree with 5 black nodes and specify a value for each node.

null or non-null



3. (10 points) What does the following section of code output?

```
public static void main(String[] Args) {
    int[] vals = {-10, -101, 98, 67, 32, 12, 58, -22, 39, 5};
    PriorityQueue<Struct> pq = new PriorityQueue<Struct>();
    for (int i = 0; i < vals.length; i++)
        pq.add(new Struct(vals[i]));
    while (!pq.isEmpty())
        System.out.println(pq.poll().value);
}
public static class Struct implements Comparable<Struct> {
    int value;
    Struct(int x) {
        value = x;
    }
    public int compareTo(Struct o) {
        if ((value & 1) == (o.value & 1))
            return value - o.value;
        return (value & 1) - (o.value & 1);
    }
}
```

-22
-10
12
32
58
98
-101
5
39
67

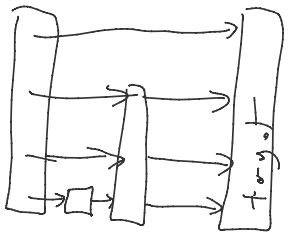
-10, 98, 32, 12, 58, -22
-101, 67, 39, 5

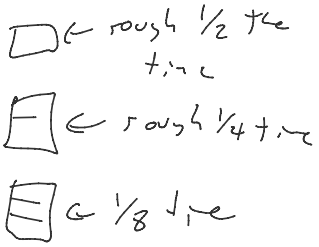
tie breaking least method to stay stable

actual sort

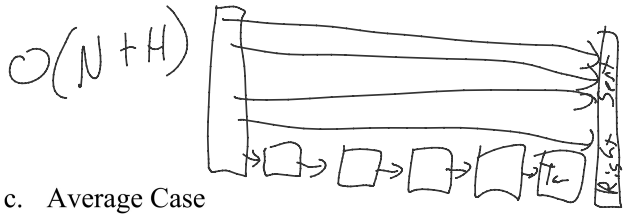
even values will have a smaller (&1)

4. (10 points) What is the best, the worst, and the average case runtime for a Skip's contains function, where the skip list selects heights for inserted values randomly with p = ½.
   a. Best Case

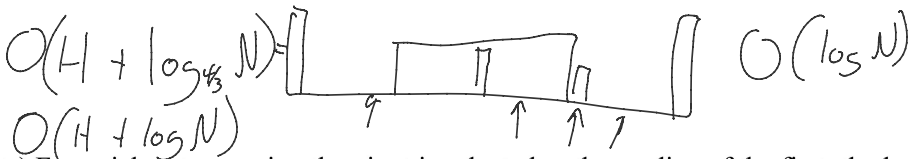




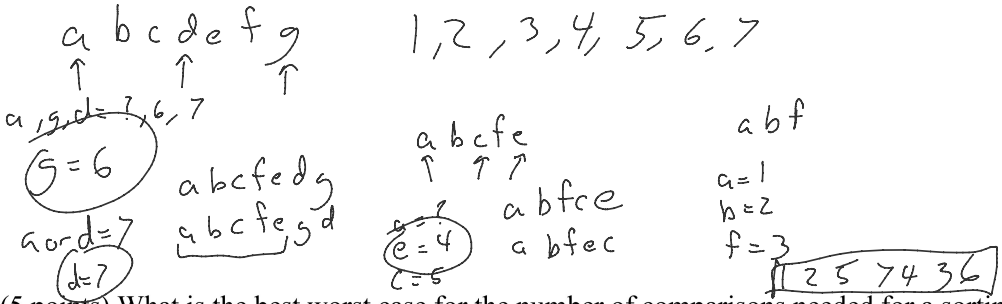
   b. Worst Case



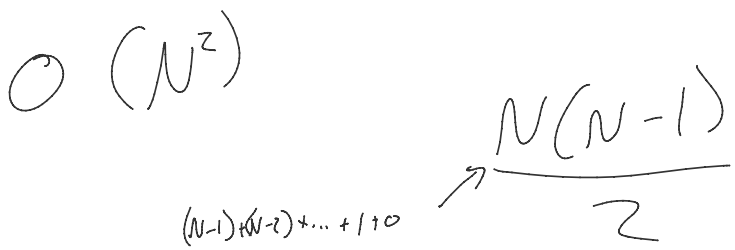
   c. Average Case



5. (10 points) For quicksort assuming the pivot is selected as the median of the first, the last, and the middle (rounded down in even length arrays) element of a list of numbers please draw an array of length 7 that has the worst case number of comparisons. **[HINT: Remember the pivot is moved to the back of the array when splitting the array]**



6. (5 points) What is the best worst case for the number of comparisons needed for a sorting method that does adjacent swaps on an array with size N?



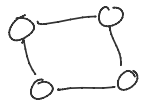
7. (5 points) List a sorting method discussed in class that has the potential to have a runtime better than O(N log(N)) worst case runtime regardless of the ordering of elements, where N is the number of elements.

Bucket

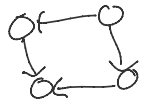
Radix

"Counting" Sort

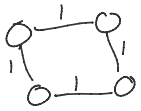
8. (8 points) Draw an example each with 4 nodes and 4 edges of graphs that are,
   a. Undirected and unweighted



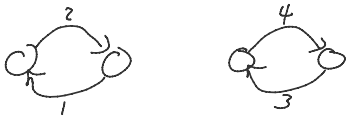
   b. Directed and unweighted



   c. Undirected and weighted



   d. Directed and weighted

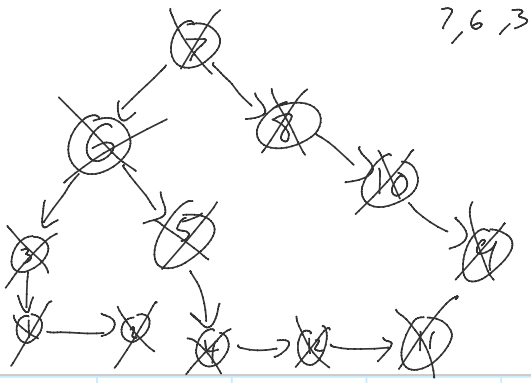


Lowest id tie break

9. (10 points) For the following undirected graph determine the order in which the nodes will be visited using a DFS starting at Node 7. The edges of the graph are the following,

(1,2)	(1,3)	(2,3)	(3,6)	(4,5)
(4,12)	(5,6)	(5,11)	(5,12)	(6,7)
(7,8)	(7,9)	(8,10)	(9,10)	(11,12)

7, 6, 3, 1, 2, 5, 9, 12, 11, 8, 10, 9

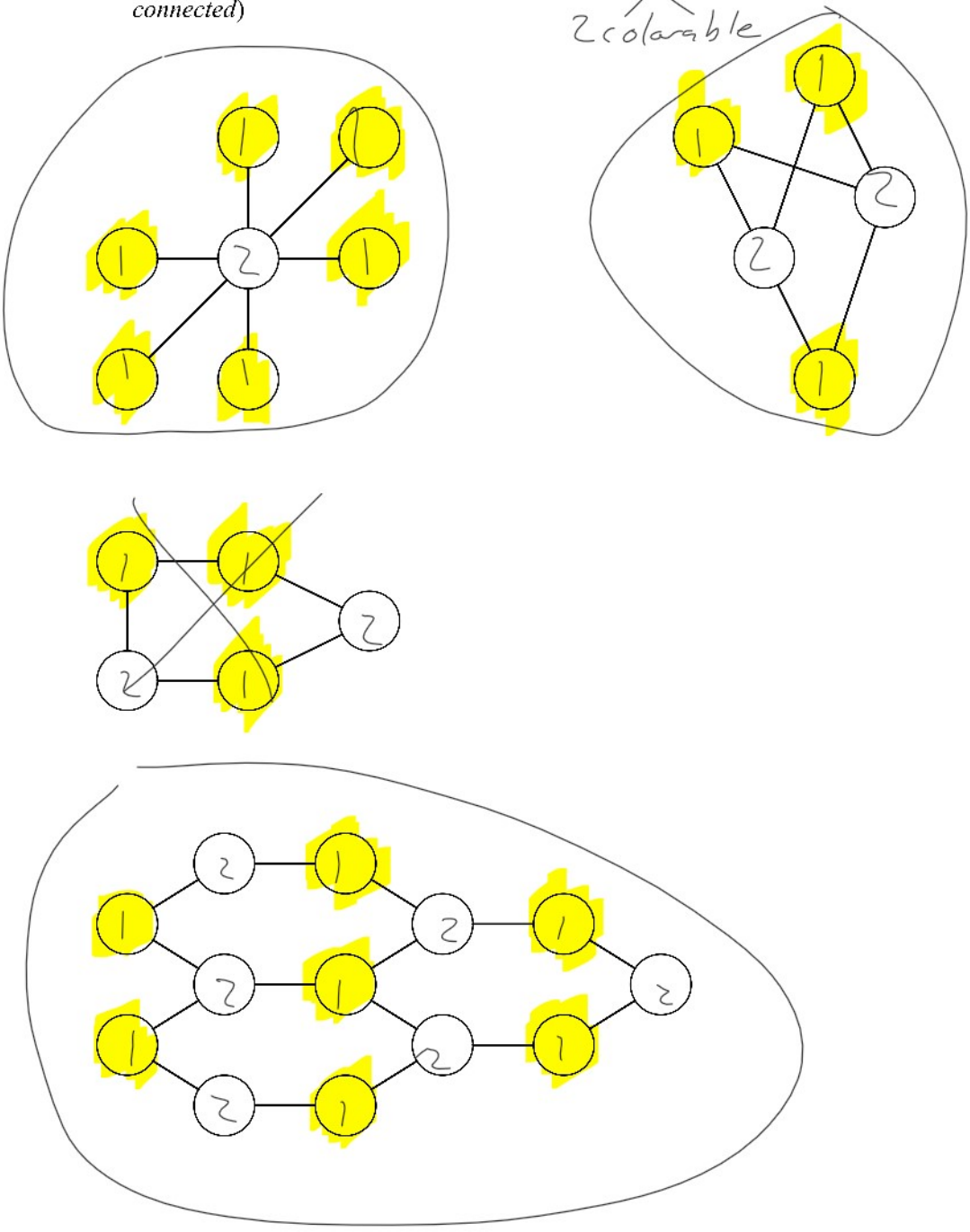


oops ha ha

1. (5 points) If you are given a graph that is *connected*, what do you know about said graph?

From any node to any other node there is at least one path

2. (8 points) Circle the following graphs that are ~~bipartite~~ 2colorable? (assume each graph is *connected*)



10. (5 points) What does DAG stand for?

Directed
Acyclic
Graph

not DAG
probably a Norbert

11. (20 points) You are put in charge of buying and selling different grades of petroleum. Unfortunately, to transfer N cubic liters of a petroleum mixture ***i*** you need to purchase a permit that costs a_i (N^2) dollars. As it turns out you can sell N liters of petroleum ***i*** for $b_i(V)$ dollars. Assuming you can only transfer K liters, determine the maximum amount of dollars you can earn.

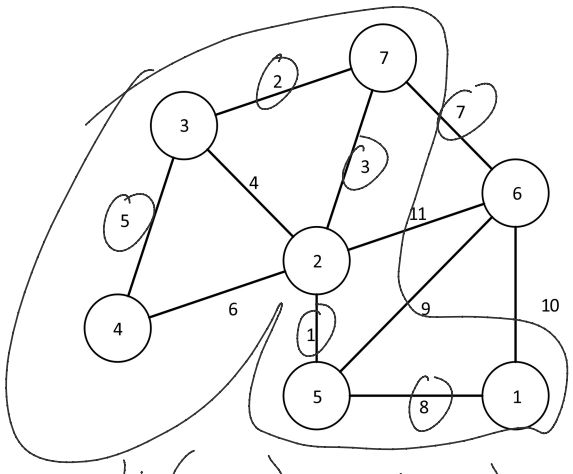
$b_i C_i - a_i(C_i)^2 \rightarrow (b_i - a_i c_i) C_i$ = Profit for C_i value

C_i is the amount of liter I take for petroleum i

Search for the Δp

Binary for the derivative of your profit graphs then greedily assign counts. See if you violated the capacity!

12. (10 points) What is the order of the edges added to the MST using the Prim's algorithm? (Start at vertex 1, and list the edges as with the ordered pairs of the incident nodes)



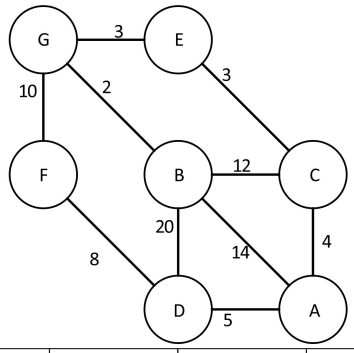
1: (1, 5) 5: (3, 4)

2: (2, 5) 6: (6, 7)

3: (2, 7) 26?

4: (3, 7)

13. (15 points) Trace through a Dijkstra's algorithm on the following graph starting a vertex D. After which list out the minimum length path (including the intermediate nodes) from D to each vertex.



Node to add

	A	B	C	D	E	F	G
D	5	20	∞	0	∞	8	∞
A	5	19	9	0	∞	8	∞
F	5	19	9	0	∞	8	18
C	5	19	9	0	12	8	18
E	5	17	9	0	12	8	15
G	5	17	9	0	12	8	15
B							

Shortest paths from D to

A DA
B DACEGB
C DAC
D D
E DACE
F DF
G DACEG

14. (5 points) What was the runtime for the Karatsuba algorithm (for multiplying numbers) discussed in class? (You can use the closed form or the recurrence relation).

$N^{\log_2 3}$

recurrence → $T(N) = 3T(N/2) + O(N)$

$x = x_0 + x_1 \cdot b$

$y = y_0 + y_1 \cdot b$

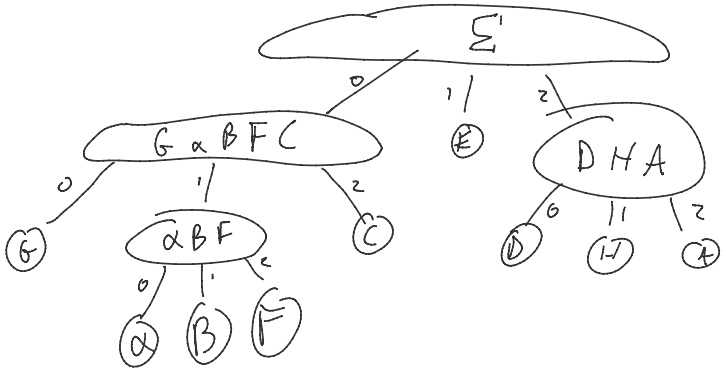
15. (10 points)* What does the Huffman Tree with an encoding grammar size of 3 look like for the following frequency table. (Use the characters 0, 1, and 2 for the grammar, where 0 is a left child 1 is the middle child and 2 is the right child. Lowest to highest frequency is assigned 0 to 2 respectively tie break is decided by the lexicographical order of the leaf nodes from left to right.)

Letter	Frequency
A	8
B	1
C	3
D	4
E	13
F	2
G	2
H	6

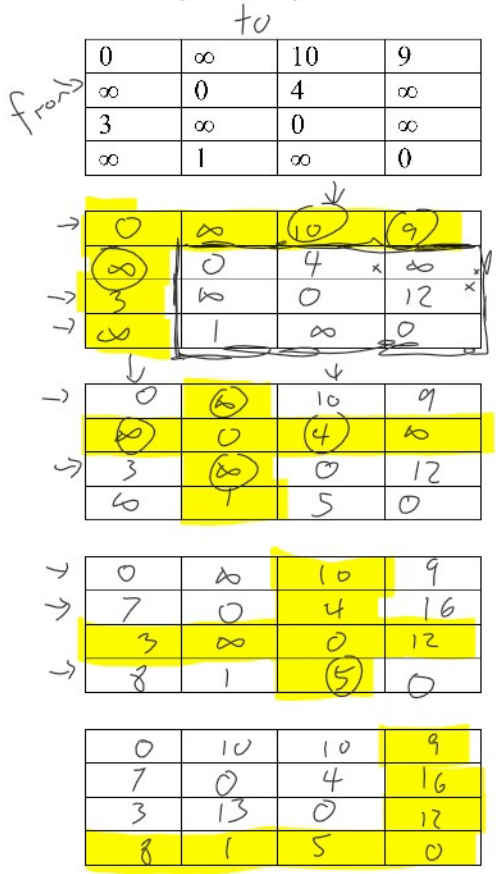
G α B F C 8

E 13

D H A 18



16. (15 points)* Trace the Floyd's algorithm on the following Graph. Show the graph after each stage of using an intermediate node.



17. Suppose you have a bag with capacity 13.

a. (10 points) What is the value of the best 0-1 selection for the following set of items? (for full points fill in the following trace table)

Name	Weight	Value
Wallet	2	8
Tablet	5	7
Computer	11	17
Paper	3	6
Pens	1	2
Umbrella	7	10

Trace Table

Cap	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Item	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wallet														
Tablet														
Computer														
Paper														
Pen														
~~Glasses~~														

b. (5 points) What is a set of items that fits in the 13 knapsack and has a maximal value?

18. (10 points) I have a set of ***N*** books I wish to check out from the library. I can only checkout each book once, and each book has a particular weight w_i. I can only carry ***W*** pounds of books, and the library won't let me check out more than ***K (N>K)*** books (I may have some overdue books). Please help me determine how many ways I can check out exactly ***K*** books. Describe (or code) an algorithm that can efficiently solve such a problem. What is your algorithms runtime?