

CS 5/7350 - Test#1
March 17, 2021

Name: Bingying Liang
ID: 48999397

1. [8 pts] You run different programs for various values of “n” and create 4 tables of the runtimes. Give the Asymptotic bounds that each of the tables support?

a.	n	time(ms)	b.	n	time(ms)	c.	n	time(ms)	d.	n	time(ms)
	10	34290		10	17		1000	3247		1000	3247
	20	512980		11	34		2000	4557		2000	3447
	30	2540070		12	68		3000	5562		4000	3647
	40	7939560		13	136		4000	6410		8000	3847
	50	19255450		14	272		5000	7156		16000	4047
	60	39751740		15	544		6000	7831		32000	4247
	70	73412430		16	1088		7000	8452		64000	4447
	80	124941520		17	2176		8000	9029		128000	4647
	90	199763010		18	4352		9000	9572		256000	4847
	100	304020900		19	8704		10000	10085		512000	5047

Solution: (a) $\Theta(n^4)$

(b) $\Theta(2^n)$

(c) $\Theta(\sqrt{n})$

(d) $\Theta(\log(n))$

2. [6 pts] Consider two different algorithms that each solve a different problem.

- Algorithm X solves Problem P_x and Algorithm X is $\Theta(n_2)$
- Algorithm Y solves Problem P_y and Algorithm Y is $\Theta(n_3)$

Determine if each of these “Yes it is true”, “Maybe it is true but doesn’t have to be”, or “No it is not true”

(a) Problem P_x is easier than Problem P_y : M

(b) Problem P_y is easier than Problem P_x : M

(c) Algorithm X is easier than Algorithm Y: Y

(d) Algorithm X is $\Omega(n^2)$: Y

(e) Algorithm X is $\omega(n^2)$: N

(f) Algorithm Y is $\Omega(n^2)$: Y

3. [5 pts] Given that $M > 100$ and $7_{31121} \bmod M = 9$ and $A = 1, 2, 3, 4$ and $B = A, B, C, D$

(a) $7^{31122} \bmod M = 63$

(b) $7^{62242} \bmod M = 81$

(c) $\frac{1}{3} \bmod 11 = 4$

(d) $|A \times B| = 16$

(e) $2^{125} \bmod 11 = 10$

4. [8 pts] How many bits of entropy are in the following messages?

(a) A message that contains 20 H's and 80 T's?

Solution: 72.19 bits

(b) A message that contains 30 H's and 70 T's?

Solution: 88.129 bits

(c) A message that contains 0 H's and 100 T's?

Solution: 0 bits

(d) A message that contains 50 H's and 50 T's?

Solution: 100 bits

5. [5 pts] Define Algorithm

Solution: A step by step procedure for solving a problem in a finite amount of time.

6. [8 pts] A 100 character message containing only A, B, C and D's is Huffman encoded. For each of the following codes, determine if they are valid. If they are not, explain why they are not and if they are valid, give a number of A, B, C and Ds that could generate that code:

(a) A = 00; B = 01; C = 10; D = 11

Solution: Valid, A: 25, B: 25, C: 25, D: 25

(b) A = 1; B = 01; C = 00; D = 001

Solution: Invalid, 001 could be "CA" or "D"

(c) A = 1; B = 01; C = 000; D = 001

Solution: Valid, A: 50, B: 30, C:10, D:10

(d) A = 1; B = 0; C = 10; D= 11

Solution: Invalid, 10 could be AB or C

7. [6 pts] Argue that the problem, S, of sorting an unsorted array of integers is at least as hard as and possibly harder than the problem, H, of creating a MIN_HEAP of that array.

Solution: Since the solution to the problem of sorting an array can be used for creating a Min-Heap (because a sorted array is a min-heap), Problem S must be just as hard or possible harder than problem H.

8. [6 pts] Draw all the non-isomorphic trees on 5 vertices.

Solution:



9. [6 pts] We want to play a game called guess an integer “n”. I will think of an integer. You can offer guesses and I’ll tell you “higher” or “lower”. Give an algorithm for guessing n that requires $\Theta(\lg(n))$ guesses. (Hint, $\Theta(\lg(n))$ could be $2\lg(n)$ or $3\lg(n)$ – To answer a common question, there is no upper bound, but the number of guesses is allowed to grow based on the size of the chosen number)

Solution:

Guess = 1

While (ASK is Guess < answer == yes) {Guess = Guess * 2}

Binary search for answer between Guess and Guess/2

10. [6 pts] You are designing a Diffie-Hellman key exchange. You choose the base of 3. Would 7, 11 or 13 make a better modulus for that base? Why?

Solution: 7 is better since it cycles through more values before running to 1.

$$3^1 \bmod 7 = 3$$

$$3^2 \bmod 7 = 2$$

$$3^3 \bmod 7 = 6$$

$$3^4 \bmod 7 = 4$$

$$3^5 \bmod 7 = 5$$

$$3^6 \bmod 7 = 1$$

$$3^7 \bmod 7 = 3$$

$$3^1 \bmod 11 = 3$$

$$3^2 \bmod 11 = 9$$

$$3^3 \bmod 11 = 5$$

$$3^4 \bmod 11 = 4$$

$$3^5 \bmod 11 = 1$$

$$3^6 \bmod 11 = 3$$

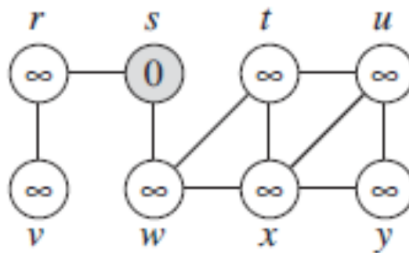
$$3^1 \bmod 13 = 3$$

$$3^2 \bmod 13 = 9$$

$$3^3 \bmod 13 = 1$$

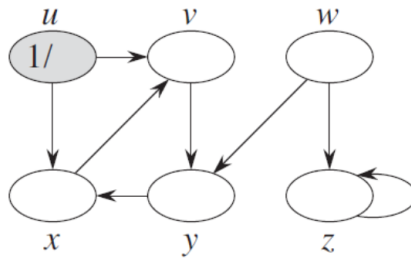
$$3^4 \bmod 13 = 3$$

11. [7 pts] Consider performing a Breadth First Search of the following graph starting at S. Give the order the vertices are popped from the queue:

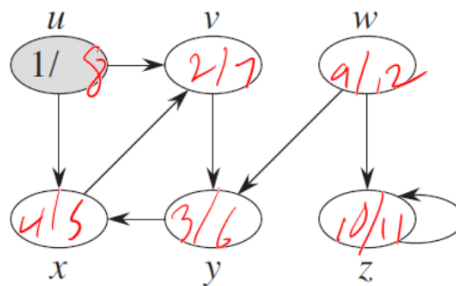


Solution: SRWVTXUY

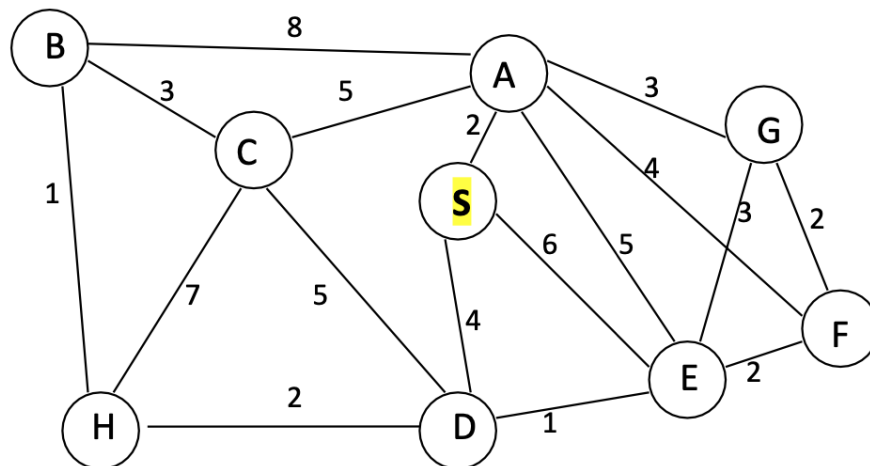
12. [7 pts] Give the discover and finalize time for each vertex of the following graph when performing a Depth First Search starting at U.



Solution:



13. [8 pts] Consider the following graph. For any questions needing a starting vertex, use vertex S as the starting vertex.



- (a) What is the value of the third edge chosen when finding a minimum spanning tree using Prim's algorithm?

Solution: 2

- (b) What is the value of the third edge chosen when finding a minimum spanning tree using Kruskal's algorithm?

Solution: 2

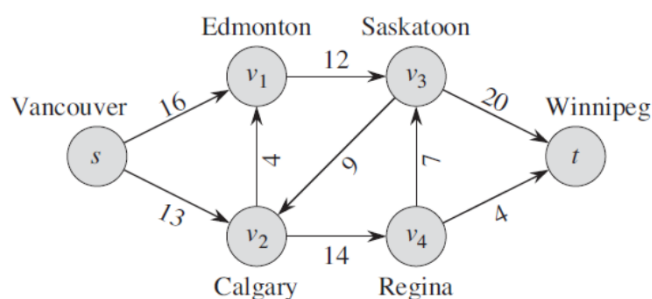
- (c) When using Dijkstra's algorithm to find the shortest path from S to all vertices, what is the order in which the vertices are reached?

Solution: S A D E G F H B C

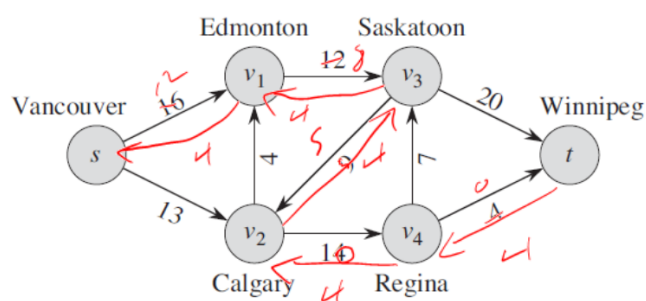
- (d) What is the weight of the minimum spanning tree?

Solution: 16

- 8 pts The graph below represents containers that are transported between these cities each day. You are determining the maximum flow from vertex S, Vancouver, to vertex T, Winnipeg, using the Ford-Fulkerson algorithm in the graph below. A Breadth First Search finds the path $s \rightarrow v_1 \rightarrow v_3 \rightarrow v_2 \rightarrow v_4 \rightarrow t$. What does the new graph look like after "removing" this flow?



Solution:



14. [6 pts] A particular algorithm on a computer requires 2 seconds to process 200 items and is $\Theta(n^3)$. You want to process 4000 items. You have a choice to either use a computer that is 10 times faster (allowing it to process 200 items in 0.2 seconds) or use the same computer with a different algorithm that still processes 200 items in 2 seconds, but has a growth rate that is $\Theta(n^2)$.

- (a) Which is the faster choice for 4000 items?

Solution: Different algorithm

(b) For what input sizes is the faster computer better?

Solution: size < 2000

(c) For what input sizes is the $\Theta(n^2)$ algorithm better?

Solution: size > 2000