

CS 5/7350 - Test#1
October 5, 2022

Name: Bingying Liang
ID: 48999397

1. [6 pts] Argue that the problem, S, of sorting an unsorted array of integers of length greater than 100 elements is at least as hard- and maybe even harder - within a constant factor of $O(1)$ than the problem, L, of finding the three largest elements of the same unsorted array of integers.

Solution: Since a solver for S can be used to solve L by sorting the array and then printing the last 3 elements, problem S must be just as hard or possible harder than Problem L.

2. [6 pts] What is an algorithm?

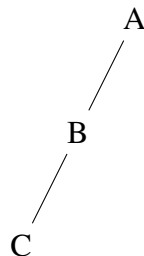
Solution: Informally, an algorithm is any well-defined computational procedure that takes some value, or set of values, as input and produces some value, or set of values, as output in a finite amount of time. An algorithm is thus a sequence of computational steps that transform the input into the output. An algorithm can be viewed as a tool for solving a well-specified computational problem. The statement of the problem specifies in general terms the desired input/output relationship for problem instances, typically of arbitrarily large size. The algorithm describes a specific computational procedure for achieving that input/output relationship for all problem instances.

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

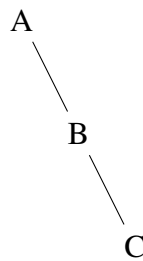
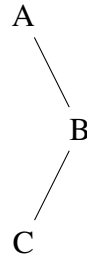
3. [5 pts] Draw two different trees where both trees have a pre order traversal of ABC and a post order traversal of CBA.

Solution:

(a)



(b)



4. [5 pts] Using n_0 equal to 10, show that $f(n) = 8n^2 + 5n + 1$ is $\Omega(n)$

Solution:

$$0 \leq c_1 n \leq 8n^2 + 5n + 1, \forall n \geq 10$$

$$0 \leq c_1 \leq 8n + 5 + \frac{1}{n}, \forall n \geq 10$$

c_1 can be 80.

Note 80 is not the tightest value, but is always less than $8n + 5 + \frac{1}{n}$

5. [8 pts] You run different programs for various values of "n" and create 4 tables of 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	1348		10	5698		100	52		4	1
	20	8348		20	5698		200	73		5	5
	30	27348		30	5698		300	89		6	30
	40	64348		40	5698		400	102		7	210
	50	125348		50	5698		500	114		8	1680
	60	216348		60	5698		600	124		9	15120
	70	343348		70	5698		700	134		10	151200
	80	512348		80	5698		800	143		11	1663200
	90	729348		90	5698		900	152		12	19958400
	100	1000348		100	5698		1000	160		13	259459200

Solution: (a) $\Theta(n^3)$

$$\frac{n_2}{n_1} = \frac{80}{40} = 2, \frac{f(n_2)}{f(n_1)} = \frac{512346}{64348} \approx 7.96 \approx 8 = 2^3$$

(b) $\Theta(1)$

$$\frac{n_2}{n_1} = \frac{80}{40} = 2, \frac{f(n_2)}{f(n_1)} = \frac{5698}{5698} = 1$$

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

$$\frac{n_2}{n_1} = \frac{800}{400} = 2, \frac{f(n_2)}{f(n_1)} = \frac{143}{102} \approx 1.401960784 \approx 1.4 \approx \sqrt{2}$$
$$\frac{n_4}{n_3} = \frac{900}{300} = 3, \frac{f(n_4)}{f(n_3)} = \frac{152}{89} \approx 1.707865169 \approx 1.7 \approx \sqrt{3}$$

(d) $\Theta(n!)$

$$\frac{f(n_2)}{f(n_1)} = \frac{f(8)}{f(7)} = 8 = \frac{8!}{7!}$$

6. [5 pts] Two people need to establish a secret key for encrypting communications. They agree to use a Diffie-Hellman key exchange with a modulus of 11 and decide on 2 as the base. Person A choose a random value performs the appropriate computations and sends the value 5 to person B. Person B chooses a random value of 3 and performs the appropriate computations:

(a) What is the value Person B sends to Person A

Solution:

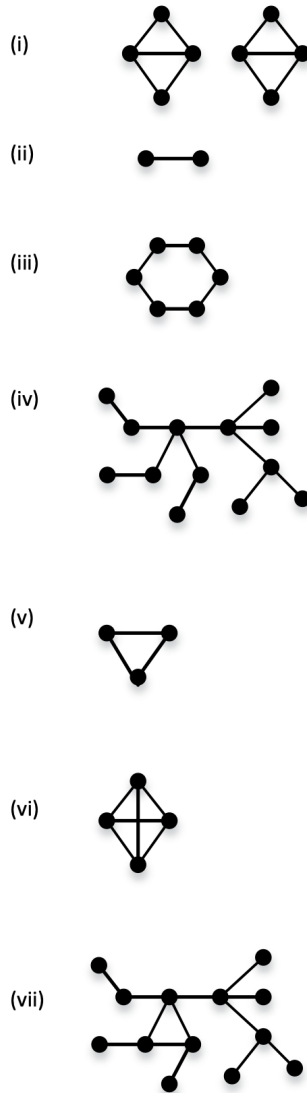
$$\therefore A \rightarrow B : 5$$
$$\therefore A \leftarrow B : 2^3 \bmod 11 = 8 \bmod 11 = 8$$

(b) What is the shared secret key between Person A and Person B

Solution:

$$a^3 \bmod 11 = 5^3 \bmod 11 = 125 \bmod 11 = 4$$

7. [7 pts] For the following graphs, indicate whether they are a tree, a complete graph, a cycle and/or are bipartite or none of the above



Solution:

- i **None**
- ii Tree, complete graph, no cycle, bipartite
- iii Not Tree, not complete graph, cycle, bipartite
- iv Tree, not complete graph, no cycle, bipartite
- v Not Tree, complete graph, cycle, not bipartite
- vi Not Tree, complete graph, **no cycle**, not bipartite
- vii **None**

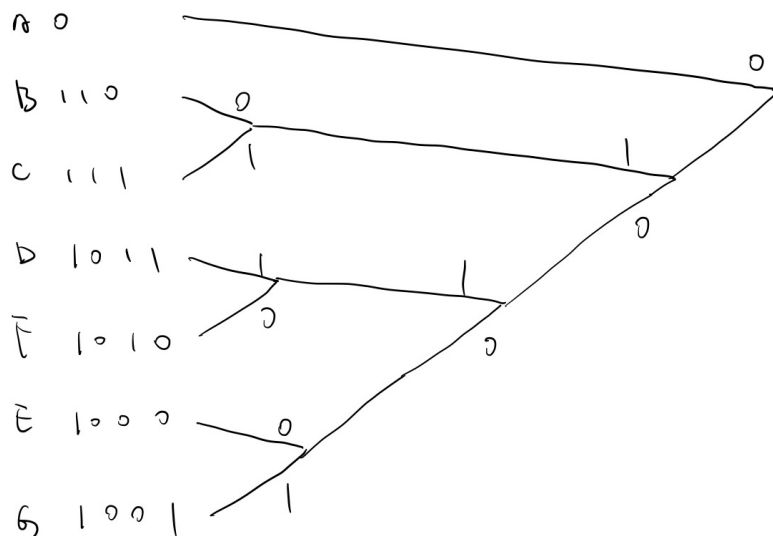
A cycle is closed walk where no vertex appears twice (except beginning and end, at least edge = points

8. [9 pts] You had a message that was 11 characters long. It was created from the letters A, B, C, D, E, F and G. You Huffman compressed the message. As a result of your compression, each letter had the following bit patterns:

- A - 0
- B - 110
- C - 111
- D - 1011
- E - 1000
- F - 1010
- G - 1001

(a) Draw the Tree that produced this compression

Solution:



(b) How many of each letter were present in the original message?

$A : 3, B : 2, C : 2, D : 1, E : 1, F : 1, G : 1$

$A : 4, B : 1, C : 2, D : 1, E : 1, F : 1, G : 1$

(c) How much entropy was present in the original message?

$$\begin{aligned}
 & 3 \log_2 \frac{1}{p_A} + 4 \log_2 \frac{1}{p_B} + 4 \log_2 \frac{1}{p_D} \\
 &= 3 \log_2 \frac{11}{3} + 4 \log_2 \frac{11}{2} + 4 \log_2 \frac{11}{1} \\
 &= 3 \times 1.874469118 + 4 \times 2.459431619 + 4 \times 3.459431619 \\
 &= 5.623407354 + 9.837726476 + 13.83772648 \\
 &= 29.29886031 \text{ bits}
 \end{aligned}$$

other may works too

$$\begin{aligned}
 & 4 \log_2 \frac{1}{p_A} + 1 \log_2 \frac{1}{p_B} + 2 \log_2 \frac{1}{p_C} + 4 \log_2 \frac{1}{p_D} \\
 &= 4 \log_2 \frac{11}{4} + 1 \log_2 \frac{11}{1} + 2 \log_2 \frac{11}{2} + 4 \log_2 \frac{11}{1} \\
 &= 28.0537481 \text{ bits}
 \end{aligned}$$

9. [6 pts] A particular algorithm on a computer requires 2 seconds to process 400 items and is $\Theta(n^3)$. You want to process 8000 items. You have a choice to either use a computer that is 10 times faster (allowing it to process 400 items in 0.2 seconds) or use the same computer with a different algorithm that still processes 400 items in 2 seconds, but has a growth rate that is $\Theta(n^2)$

(a) Which is the faster choice for 8000 items?

Solution:

$$\text{faster computer: } time = \left(\frac{8000}{400}\right)^3 \times 0.2 = (20)^3 \times 0.2 = 1600 \text{ seconds}$$

$$\Theta(n^2) \text{ algorithm: } time = \left(\frac{8000}{400}\right)^2 \times 2 = (20)^2 \times 2 = 800 \text{ seconds}$$

$\therefore \Theta(n^2)$ algorithm better for 8000 items

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

Solution:

suppose input sizes is n

$$\left(\frac{n}{400}\right)^3 \times 0.2 < \left(\frac{n}{400}\right)^2 \times 2 \rightarrow n < 4000$$

\therefore when the input sizes is smaller than 4000

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

Solution:

When the input sizes is larger than 4000.

10. [9 pts] Consider two different algorithms that each solve a different problem.

- Algorithm X solves Problem P_x and Algorithm X is $O(n^3)$ and $\Omega(n)$
- Algorithm Y solves Problem P_y and Algorithm Y is $O(n^4)$ and $\Omega(n^2)$
- Algorithm Z solves Problem P_z and Algorithm Z is $\Theta(n^5)$

From the information above, 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_y is harder than Problem P_z : M

(b) Problem P_z is harder than Problem P_x : M

(c) Algorithm Y is harder than Algorithm X : **M**

Solution: Because the Algorithm X is between $O(n^3)$ and $\Omega(n)$, Algorithm Y is between $O(n^4)$ and $\Omega(n^2)$. They can be n^3 at the same time, which means may be.

(d) Problem Y is $\Omega(n^2)$: **M**

Solution: Because the Algorithm Y is $O(n^4)$ and $\Omega(n^2)$, the problem the worst must be equal or less than Algorithm. Therefore, the Problem Y is $\Omega(n^2)$ or $\Omega(n)$...

(e) Problem X is $\omega(\log n)$: M

(f) Problem Y is $O(n^4)$: Y

(g) Problem X is $o(n^4)$: **Y**

Solution: Because the Algorithm X is $O(n^3)$ and $\Omega(n)$, therefore, the problem X must equal or less $O(n^3)$. Therefore, if $o(n^4)$ is ok.

(h) Algorithm Z is $\Omega(n)$: Y

(i) Algorithm Z is $w(n)$: Y

11. [10 pts] Setup the table as shown in class and determine 1/10 modulo 7657.

Solution:

mod 7657	1	2	3	4	5	6	7
7657	$7657 + 1 = 7658$	$2 \times 7657 + 1 = 15315$	22972	30629	38286	45943	53600

$$(53600) \bmod 7657 = (5360 \times 10) \bmod 7657 = 1$$

$$\left(\frac{1}{10} \times 10\right) \bmod 7657 = 1$$

$$\left(\frac{1}{10}\right) \bmod 7657 = 5360$$

12. [6 pts] Solve the following for the value of A. Ensure your answer is an integer between 1 and 3058 inclusive.

$$1822 = ((5 * A) + 89) \text{ modulo } 3059$$

Solution:

$$((5 * A) + 89) \bmod 3059 = 1822$$

$$(5 * A) \bmod 3059 = 1822 - 89$$

$$(5 * A) \bmod 3059 = 1733$$

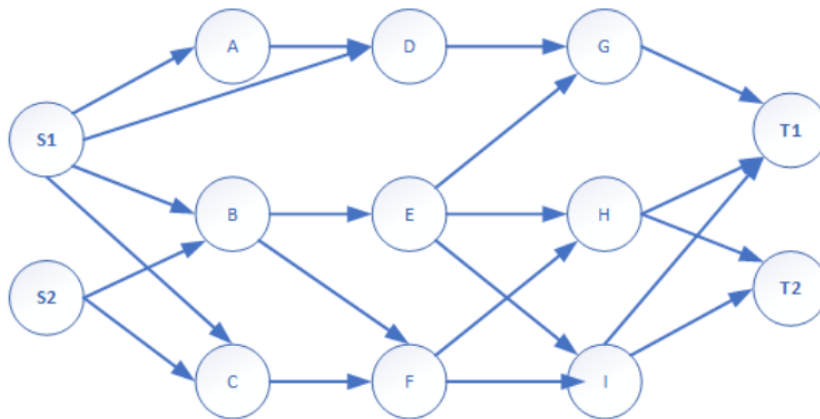
$$\therefore (5 * A) = B \times 3059 + 1733$$

$$A = \frac{B \times 3059 + 1733}{5}$$

\therefore A is integer, and B is also integer

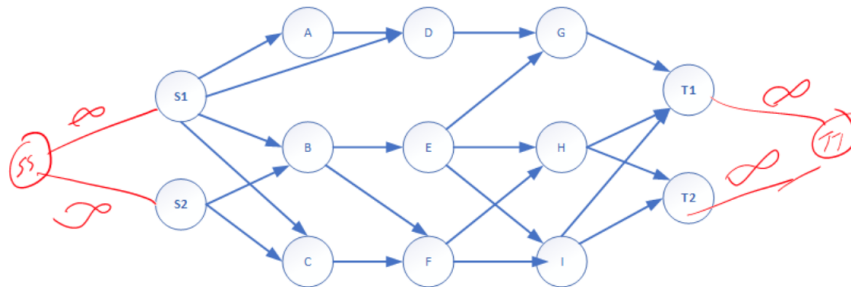
$$\therefore A = 2182$$

13. [6 pts] You have the following graph:



Assume each of the edges have numbers associated with them that represent the flow of goods. You can think of S1 and S2 as being warehouses that need to be emptied and you can think of T1 and T2 as being storage facilities that can store the items from the S1 and S2 warehouses which are being emptied. Items leaving the S1 and S2 warehouses can be split between the T1 and T2 storage facilities. That is, you do not care which storage facility T1 or T2 each item arrives at. You just want to empty them from S1 and S2 as fast as possible.

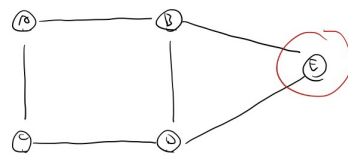
How would you modify the graph above to allow you to use the Ford-Fulkerson maximum flow algorithm we learned in class that works for a single source, S, and a single sink, T, to solve this problem?



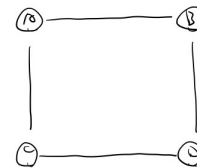
Solution:

14. [6 pts] Draw a graph and give a smallest last vertex ordering where the terminal clique is not the largest complete subgraph. Circle the first vertex you write down in your smallest last ordering.

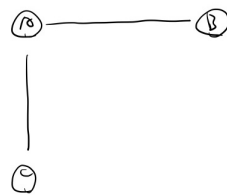
Solution:



(1)



(2)



(3)



(4)



(5)

A	B	C	D	E
0	1	1	2	2
C	A	B	D	E

Complete graph

largest complete subgraph is B, D, E.

15. [6 pts] Answer the following questions:

- (a) What is the weight of a minimum spanning tree for a connected, acyclic graph with $|V|$ vertices where all edges have a weight of 4?

Solution: $4(|V| - 1)$

- (b) What is the weight of a minimum spanning tree for a cycle with 10 vertices where all edges have a weight of 4?

Solution: $4(|V| - 1) = 4 \times (10 - 1) = 36$

- (c) A complete bi-partite graph $B_{j,k}$ is a graph which has J vertices in one partition and k vertices in another partition and all possible edges present between the partitions. For which values of j and which values of k are the degrees of the vertices even?

Solution:

If j is even, then the degree K vertices are even.

If k is even, then the degree J vertices are even.

When j, k are both even.