CS 5/7350 - Test#3
May 11, 2022

Name: Bingying Liang
ID: 48999397

1. [11 pts] Consider the following NP completeness questions. Answer them with the best answer of "some" "all" "none" or "unknown"

   (a) Which Problems in NP are also in P? ("some" "all" "none" or "unknown")

   Solution: unknown

   (b) Which Problems in P are also in NP? ("some" "all" "none" or "unknown")

   Solution: all

   (c) Which Problems in NP-Hard are also in NP? ("some" "all" "none")

   Solution: some

   (d) Which Problems in NP-Complete are in NP-Hard ("some" "all" "none" or "unknown")

   Solution: all

   (e) If someone can solve an NP-Complete problem in Polynomial Time, then all NP and all NP-Hard problems can be solved in polynomial time. (true or false)

   Solution: false

   (f) If someone can solve an NP-Complete problem in Polynomial Time, then all NP and all NP-Complete problems can be solved in polynomial time. (true or false)

   Solution: true

   (g) At least 1 NP problem has a known solution to solve it in polynomial time? (True or False)

   Solution: true

   (h) All NP-Complete problems are in P ("true" "false" or "unknown")

   Solution: unknown

   (i) Which NP-Hard Problems are also NP-Complete? ("some" "all" "none" or "unknown")

   Solution: some

   (j) To show a problem, Q, is NP-Complete, you must show Problem Q is NP and that a solver for another NP-Hard problem can solve problem Q as well. (True or False)

   Solution: False

   (k) To show a problem, Q, is NP-Complete, you must show Problem Q is NP and that a solver for problem Q can solve another NP-Hard problem. (True or False)

Solution: True

2. [6 pts] Consider an LZW compression scenario with a dictionary that contained 1024 entries. In this dictionary, entries 0-255 were the standard ASCII values and entries 256-1023 were the dynamic part of the dictionary. This compression was able to compress a file of 1000kB to 750kB:

   (a) What is one reason that a larger dictionary of size 2048 with dynamic entries from 256-2047 might cause the file to compress SMALLER than 750kB?

   Solution: A larger dictionary can allow more patterns to be remembered and used without to build them again.

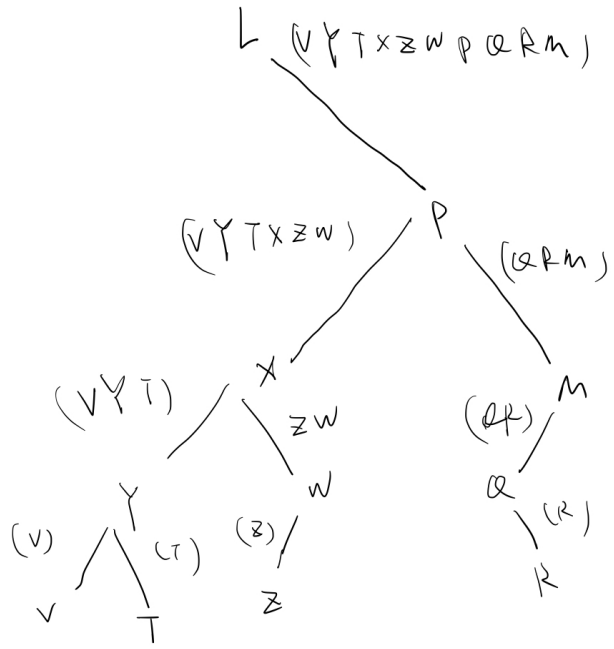
   (b) What is one reason that a larger dictionary of size 2048 with dynamic entries from 256-2047 might cause the file to compress LARGER than 750kB

   Solution: A larger dictionary means that more bits are needed for each symbol in the compressed message.

3. [6 pts] You have a tree with the following in-order and pre-order traversals. Draw the tree:

   IN ORDER: L V Y T X Z W P Q R M
   PRE_ORDER: L P X Y V T W Z M Q R

   Solution:



4. [6 pts] You have 3 dice. Each one is different.

- Die #1 has sides $\{0, 1, 2\}$ with a
- Die #2 has sides $\{1, 2, 3\}$ with a
- Die #3 has sides $\{0, 1\}$ with a

(a) Fill in the table for the dynamic programming algorithm to solve the problem.

Solution:

	Die 1,	Die 1,2	Die 1,2,3	
0	1	0	0	
1	1	1	1	
2	1	2	3	
3	0	3	5	
4	0	2	5	
5	0	1	3	
6	0	0	1	

(b) What is the probability of rolling a 0?

Solution: 0

(c) What is the probability of rolling a 3?

Solution: $\frac{5}{18}$

(d) What is the probability of rolling a 6?

Solution: $\frac{1}{18}$

5. [6 pts] Answer the following questions.:

(a) A program requires 5s to attack an encryption key of 128 bits. If the running time is $\Theta(2^n)$ about how many years would it take to brute force attack an encryption key of 256 bits? (note there are about 32 million seconds in a year)

Solution:

$$\frac{2^{256}}{2^{128}} \times 5s = 2^{128} \times 5s = 2^{128} \times 5 \times \frac{1}{32 \times 10^6}$$
$$= \frac{2^{128}}{2^5} \times 5 \times 10^{-6} = 2^{123} \times 5 \times 10^{-6}\ years$$

(b) A program requires 5s to attack an encryption key of 128 bits. If you have access to a quantum computer where the running time is $\Theta(n^2)$ about how many seconds would it take to brute force attack an encryption key of 256 bits?

Solution: 20s

6. [6 pts] Use the DGT algorithm discussed in class to determine how to represent the value 1023 using the number system β=5, D = { -2, -1, 0, 1, 7 }. Show your work

Solution:

$$
\begin{aligned}
&-2 \bmod 5 = 3, -1 \bmod 5 = 4, 7 \bmod 5 = 2 \\
&1023 \bmod 5 = 3 \rightarrow -2 \\
&1023 - (-2) = 1025 \\
&1025 \div 5 = 205 \\
&- - - - - - - - - - - - - - - - - - \\
&205 \bmod 5 = 0 \\
&205 - 0 = 205 \\
&205 \div 5 = 41 \\
&- - - - - - - - - - - - - - - - - - \\
&41 \bmod 5 = 1 \\
&41 - 1 = 40 \\
&40 \div 5 = 8 \\
&- - - - - - - - - - - - - - - - - - \\
&8 \bmod 5 = 3 \rightarrow -2 \\
&8 - (-2) = 10 \\
&10 \div 5 = 2 \\
&- - - - - - - - - - - - - - - - - - \\
&2 \bmod 5 = 2 \rightarrow 7 \\
&2 - 7 = -5 \\
&-5 \div 5 = -1 \\
&- - - - - - - - - - - - - - - - - - \\
&-1 \bmod 5 = -1 \\
&-1 - (-1) = 0 \\
\therefore\ &\bar{1}7\bar{2}10\bar{2}
\end{aligned}
$$

7. [8 pts] You have two strings, A and B.

   - String A has a length of 11.
   - String B has a length of 8.
   - String C has an unknown length.
   - The Longest Common Subsequence between String A and C is 5.

   (a) What is the minimum length of String C?

   Solution: 5

(b) What is the maximum length of String C?

Solution: infinitely

(c) What is the minimum length of the Levensthein Edit Distance of String A and String C ?

Solution: 6

(d) What is the maximum length of the Levensthein Edit Distance of String A and String B?

Solution: 11

8. [6 pts] A program takes 10 seconds to process a data set of 1000 items using an algorithm that is $\Theta(n^3)$. You want to process a data set of 10,000 items.

(a) How long would it take to process these 100,000 items on a computer that is 5 times faster using the algorithm that is $\Theta(n^3)$?

Solution: $10^6 \times 2s$

(b) How long would it take to process these 100,000 items if the computer is the same speed, but the algorithm is $\Theta(n^2)$ instead?

Solution: $10^5 s$

9. [9 pts] Compute the following. Assume Graph G has $|V|$ vertices and each edge has a weight of 'w'. Give your answers in terms of "V" and "w" as appropriate.

(a) If Graph G is a cycle, what is the maximum flow between any two vertices?

Solution: 2w

(b) If Graph G is complete, what is the maximum flow between any two vertices?

Solution: (v-1)w

(c) If Graph G is a tree, what is the maximum flow between any two vertices?

Solution: w

(d) If Graph G is a cycle, the value of the minimum spanning tree of graph G is?

Solution: (v-1)w

(e) If Graph G is complete, the value of the minimum spanning tree of graph G is?

Solution: (v-1)w

(f) If Graph G is a tree, the value of the minimum spanning tree of graph G is?

Solution: (v-1)w

(g) If Graph G is a cycle, for what values of $|V|$ does graph G have an Euler Tour?

Solution: All

(h) If Graph G is complete, for what values of $|V|$ does graph G have an Euler Tour?

Solution: $|V|$ is odd.

(i) If Graph G is a tree, for what values of $|V|$ does graph G have an Euler Tour?

Solution: None

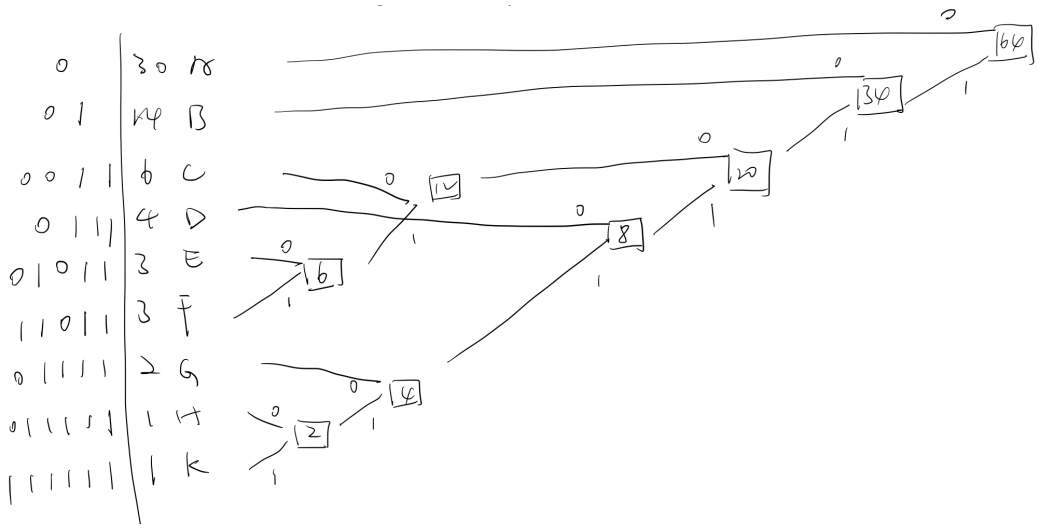
10. [5 pts] Argue that the problem, S, of sorting an unsorted array of integers of length greater than 100 elements is at least as hard - and maybe even harder - than the problem, L, of finding the median of the same array.

Solution: I can use a solver for S to solve L by sorting the array and returning the element in the middle index, since a solver for S can solve L, s must be at least as hard or possibly harder than L.

11. [9 pts] A message contains the following number of each symbol:

30 A's, 14 B's, 6 C's, 4 D's, 3 E's, 3 F's, 2 G's, 1 H and 1 K.

(a) Create a Huffman coding for each symbol:



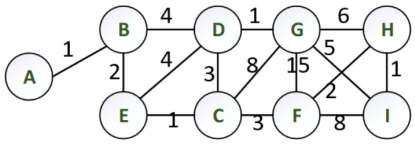
(b) How many bits are in the entire Huffman coded message?

Solution: 150 bits

(c) How much entropy does each "C" have?

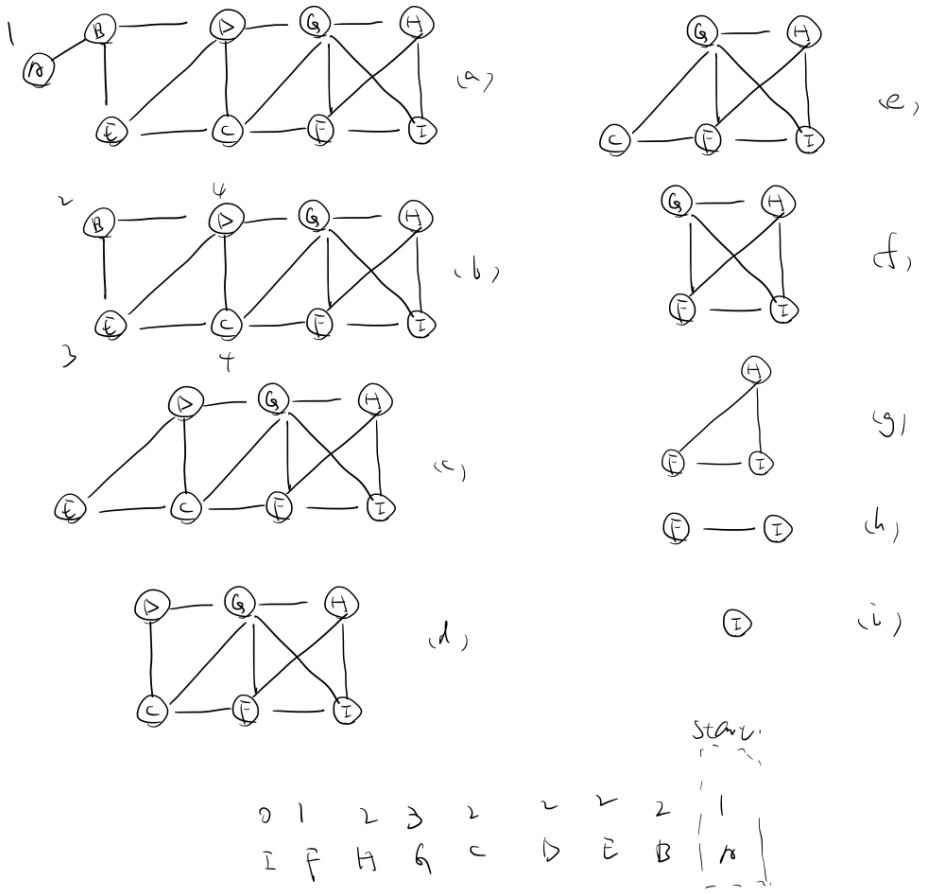
Solution: $\log_2(\frac{32}{3})$ bits

12. [6 pts] Consider the following graph. When necessary for the algorithm, use vertex C as the starting vertex:



(a) Give a smallest last vertex ordering for the graph. Circle in your ordering the first vertex you wrote down for that ordering.

Solution:



(b) What is the edge you would choose 3^{rd} when finding a minimum spanning tree with Kruskal's algorithm?

Solution: 1: EC, AB, DG, HI

(c) What is the edge you would choose 3^{rd} when finding a minimum spanning tree with Prim's algorithm?

Solution: AB

13. [4 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 chooses 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: 8

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

Solution: 4

14. [8 pts] Consider an RSA encryption system that has a public key of 339251 for the value e and 748081 for the value of the modulus N. You also saw a message that had been encrypted by the public key. The value of this encrypted message is 2.

(a) You are able to factor N=748081 into the product of two prime numbers 853 * 877. What is the value of the private key? Show your work including the table for computing the Extended Euclidean Algorithm.

Solution: d = 11; private(11, 748081)

(b) What was the original message before encryption? (Give an integer)

Solution: 2048

15. [4 pts] Using n_0 equal to 10, show that $f(n) = 6n^3 + 2n^2 + 4n + 1$ is $\Theta(n^3)$.

Solution:

$$
\begin{aligned}
\Omega(n^3) :& 0 \leq c_1 g(n) \leq f(n), \forall n = n_0 = 10 \\
& c_1 n^3 \leq 6n^3 + 2n^2 + 4n + 1, \forall n = n_0 = 10 \\
& c_1 \leq 6 + \frac{2}{n} + \frac{4}{n^2} + \frac{1}{n^3}, \forall n = n_0 = 10 \\
& c_1 \leq 6 + \frac{2}{10} + \frac{4}{100} + \frac{1}{1000} \\
& \therefore c_1 = 6 \text{ can let } f(n) \text{ is } \Omega(n^3), \forall n = n_0 = 10
\end{aligned}
$$

$$
\begin{aligned}
O(n^3) :& 0 \leq f(n) \leq c_2 g(n), \forall n = n_0 = 10 \\
& 6n^3 + 2n^2 + 4n + 1 \leq c_2 n^3, \forall n = n_0 = 10 \\
& 6 + \frac{2}{n} + \frac{4}{n^2} + \frac{1}{n^3} \leq c_2, \forall n = n_0 = 10 \\
& 6 + \frac{2}{10} + \frac{4}{100} + \frac{1}{1000} \leq c_2 \\
& 6.241 \leq c_2 \\
& \therefore c_2 = 6.241 \text{ can let } f(n) \text{ is } O(n^3), \forall n = n_0 = 10
\end{aligned}
$$