CS 5/7350 – Test 1
March 8, 2023

Name: Bingying Liang

- This exam is **closed book** and **closed notes**.
- Only the approved TI-30Xa calculator
- No cell phones, or other electronics.
- Pencil and/or pen only are permitted.
- Two Scratch Pages are on the back.
- It is **3 hours** in duration.
- You should have 15 problems. Pay attention to the point value of each problem and dedicate time as appropriate.

On my honor, I have neither given nor received unauthorized aid on this exam.

SIGNED: Bingying Liang

DATE: 03.09.2023

CS 7350 – Test #1
March 8, 2023

Name: Bingying Liang

ID: 48999397

[+5 pts CS-5350]

1. [5 pts] Circle the asymptotically larger function OR circle both if they are the same.

a. f(n) = 2n and g(n) = 8n

b. $f(n) = 2^n$ and $g(n) = 3^n$

c. f(n) = n! and $g(n) = n^n$

d. $f(n) = n^2$ and $g(n) = n^3$

e. $f(n) = \lg(n^2)$ and $g(n) = \lg(n^3)$

f. f(n) = n! and g(n) = (n+1)!

g. $f(n) = \log_{10} n$ and $g(n) = \log_2 n$

h. $f(n) = \lg(2^n)$ and g(n) = n

i. f(n) = lg(n!) and $g(n) = n \lg(n)$ or $\log n^n$

j. $f(n) = \lg(2^n)$ and $g(n) = \lg(3^n)$

2. [6 pts] Argue that the problem, S, of sorting an unsorted array of integers is at least as hard - and maybe even harder - than the problem, M, of finding the minimum element of the same unsorted array of integers.

Sol: Since we can use a solver of problem S to sorting an unsorted array of integers, and use the solver to solve problem M to find the minimum element. Therefore, problem S is as least as hard and maybe even harder than the problem M.

3. [5 pts] Using n_0 equal to 100, find the tightest C_1 and C_2 to show that $f(n) = 7n^2 + 4n + 7$ is $\Theta(n^2)$.

Sol: $\Omega(n^2)$: $0 \le c_1 g(n) \le f(n)$, $\forall n \ge n_0$

$0 \le c_1 n^2 \le 7n^2+4n+7$, $\forall n \ge n_0$

$c_1 \le 7 + \frac{4}{n} + \frac{7}{n^2}$, $\forall n \ge 100$

$c_1 \le 7 + \frac{4}{100} + \frac{7}{100^2}$

$\Rightarrow c_1 \le 7 + 0.04 + 0.0007$

≤ 7.0407

$O(n^2)$: $0 \le f(n) \le c_2 g(n)$, $\forall n \ge n_0$

$0 \le 7n^2+4n+7 \le c_2 n^2$, $\forall n \ge n_0$

$\frac{7n^2}{n^2} + \frac{4n}{n^2} + \frac{7}{n^2} \le c_2$, $\forall n \ge 100$

$7 + \frac{4}{n^2} + \frac{7}{n^2} \le c_2$

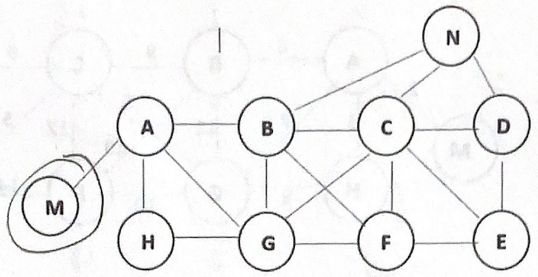
$7 + \frac{4}{100} + \frac{7}{100^2} \le c_2$

$\Rightarrow 7.0407 \le c_2$

$\therefore c_1 = c_2 = 7.0407$

$\therefore \Theta(n^2)$

4. [8 pts] Consider the following graph:



a) Give a Smallest Last Vertex Ordering for the graph where the terminal clique is the largest complete subgraph. Circle the vertex you removed first in your ordering.

0 1 2 3 1 2 2 3 2 2 1
C B G F N D E A H M

the largest complete subgraph

removed first

b) Give a Smallest Last Vertex Ordering for the graph where the terminal clique is not the largest complete subgraph. Circle the vertex you removed first in your ordering.

0 1 2 2 2 3 3 2 2 1
B C E D N B G A H M

Not the largest complete subgraph

removed first.

c) A smallest last ordering for a different graph has a terminal clique of size 15 and a largest degree when deleted of 17.

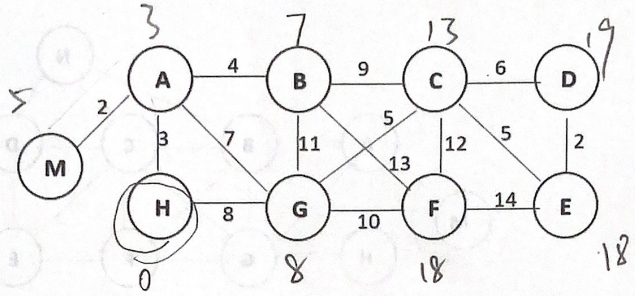
i. As an upper bound, how many colors might be needed for coloring the graph ?

18

ii. As a lower bound, how many colors must be required for coloring the graph?

16

5. [6 pts]Consider the following graph: For any algorithms below requiring a starting vertex, **use vertex H**

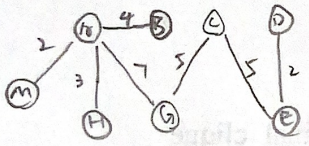


a) What is the value of the third edge chosen when computing the minimum spanning tree with Kruskal's Algorithm

2 : AM ; DE
3 : AH
AH = 3

b) What is the value of the third edge chosen when computing the minimum spanning tree with Prim's Algorithm

AB = 4



c) What is the value of the minimum spanning tree.

2+2+3+4+5+5+7 = 4+7+10+7 = 11+17 = 28

d) You want to find the shortest path from vertex H to all other vertices. What is the order you reach the other vertices using Dijkstra's Single Source Shortest Path algorithm?

0	3	5	7	8	13	18	18	19
H	A	M	B	G	C	F	E	D

6. [6 pts] Describe how you could write an algorithm which uses Dijkstra's Single Source Shortest Path algorithm as a building block to find the shortest path between all pairs of vertices in the graph above.

For each vertex in Graph:
dist = ∞
WHILE (pairs not empty)
For each neighbor
if dist < current
dist = current

If Dijkstra's Single Source Shortest Path algorithm had an asymptotically bounded running time of Θ(f(n)), what is the running time of your algorithm?

$\Theta(n^2)$

7. [15 pts] Consider three different implementations that each solve a different problem.
   - Implementation X solves Problem Px and Implementation X is $\Theta(n)$
   - Implementation Y solves Problem Py and Implementation Y is $\Theta(2^n)$
   - Implementation Z solves Problem Pz and Implementation Z is $O(n^2)$
   -

   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. Maybe Problem Py is harder than Problem Px

   b. Yes Implementation Y is harder than Implementation X

   c. Maybe Problem X is Ω (n)

   d. No Problem X is ω (n)

   e. Yes Problem Z is O (n^3)

   f. Yes Problem Z is O (n^2)

   g. Maybe Problem Y is O (n)

   h. Maybe Problem X is o (n)

   i. Yes Implementation X is Ω (n)

   j. No Implementation X is ω (n)

   k. Yes Implementation X is O (n^4)

   l. Maybe Implementation Z is O (n)

   m. Yes Implementation Z is O (n^3)

   n. Maybe Implementation Z is Ω (n)

   o. No Implementation Y is O (n)

8. [6 pts] Answer the following questions:

a) What is the maximum flow between two vertices for a complete graph with |V| vertices where all edges have a weight of w?

$(|V|-1)w$

b) What is the maximum flow between two vertices for a tree with |V| vertices where all edges have a weight of w?

w

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. What is the maximum flow between the two partitions for a complete bi-partite graph $B_{j,k}$ where all edges have a weight of 3?

$3(j \times k)$

d) What is the weight of a minimum spanning tree for a connected bi-partite graph $B_{j,k}$ where all edges have a weight of 3?

$3(j+k-1)$

9. [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 4 to person B. Person B chooses a random value of 5 and performs the appropriate computations:

a. What is the value Person B sends to Person A

$2^a \% 11 = 5$ → ← $2^5 \% 11 = 32 \% 11 = 10$

10

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

$5^5 \% 11 = 3125 \% 11 = 1$

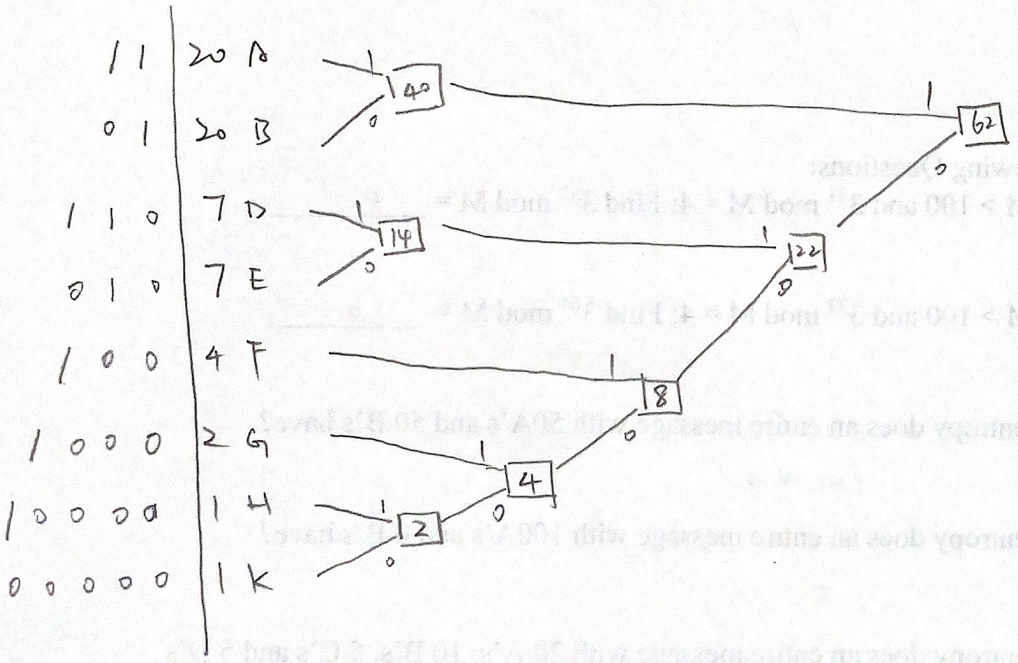
10. [8 pts] Consider a Huffman encoding of the following string.

20 A's, 20 B's, 7 D's, 7 E's, 4 F's, 2 G's, 1 H and 1 K.

How many bits are in the entire message if each symbol is encoded with 3 bits?

$(20+20+7+7+4+2+1+1)\times 3 = (40+14+6+2)\times 3 = (54+8)\times 3 = 62\times 3 = 186$ bits

Create a Huffman encoding of the bits for each symbol:



How many bits are in the entire Huffman coded message?

$20\times 2 + 20\times 2 + 7\times 3 + 7\times 3 + 4\times 3 + 2\times 4 + 1\times 5 + 1\times 5 = 40+40+21+21+12+8+5+5$

$= 80 + 42 + 20 + 10$

$= 152$ bits

How much entropy is in the entire message?

Total $= 20+20+7+7+4+2+1+1 = 62$

$P_A = \frac{20}{62} = \frac{10}{31}$ $P_B = \frac{20}{62} = \frac{10}{31}$ $P_D = \frac{7}{62}$ $P_E = \frac{7}{62}$ $P_F = \frac{4}{62} = \frac{2}{31}$ $P_G = \frac{2}{62} = \frac{1}{31}$ $P_H = \frac{1}{62}$ $P_K = \frac{1}{62}$

$\text{Entropy} = 20\log_2\frac{1}{P_A} + 20\log_2\frac{1}{P_B} + 7\log_2\frac{1}{P_D} + 7\log_2\frac{1}{P_E} + 4\log_2\frac{1}{P_F} + 2\log_2\frac{1}{P_G} + \log_2\frac{1}{P_H} + \log_2\frac{1}{P_K}$

$= 40\log_2\frac{31}{10} + 14\log_2\frac{62}{7} + 4\log_2\frac{31}{2} + 2\log_2 31 + 2\log_2 62$

$= 65.29072862 + 44.05577944 + 15.81678524 + 9.908392621 + 11.90839262 \approx 146.98$ bits

$\Phi(p \cdot q) = (p-1)(q-1)$

$\Phi(3 \times 5) = \Phi(15) = 8$

11. [6 pts] Answer the following Questions:

a. Compute $\Phi(31*29)$ 840

b. Compute $11^{\Phi(35879)} \% 35879$ 1

c. Compute $11^{\Phi(35879)+1} \% 35879$ 11

12. [7 pts] Answer the following Questions:

a. Given that $M > 100$ and $3^{31} \bmod M = 4$; Find $3^{32} \bmod M =$ 12

$3^{31} \cdot 3^1 \quad 4 \times 3$

b. Given that $M > 100$ and $3^{32} \bmod M = 4$; Find $3^{64} \bmod M =$ 16

$3^{32} \cdot 3^{32}$

c. How much entropy does an entire message with 50A's and 50 B's have?

100 bits

d. How much entropy does an entire message with 100A's and 0 B's have?

0 bits

$20+10+5+5 = 30+10 = 40$

e. How much entropy does an entire message with 20 A's, 10 B's, 5 C's and 5 D's have?

$P_A = \frac{20}{40} = \frac{1}{2}$ $P_C = \frac{5}{40} = \frac{1}{8}$ $P_D = \frac{1}{8}$ $P_B = \frac{10}{40} = \frac{1}{4}$

$20 \log_2 2 + 10 \times \log_2 4 + 5 \times \log_2 8 + 5 \log_2 8$

$= 20 + 10 \times 2 + 5 \times 3 = 20 + 20 + 15 = 55$ bits

f. Compute - 7 mod 11 4

g. Compute (½) mod 7 4

$\frac{1}{2} \times 2 \bmod 7 = 1 \quad 2 \times 4 \bmod 7$

$8 \bmod 7 = 1$

h. Compute - (½) mod 13 6

$-\frac{1}{2} \times 2 \bmod 13 =$

$-1 \bmod 13 = 12$

$2 \times 6 \bmod 13 = 12 \bmod 13 =$

13. [6 pts] What is an algorithm?

An algorithm is step by step procedure to solve a problem in finite amount of time.

14. [6 pts] Answer the following questions.:

n t
64 9

a) A program requires 9 days to brute force attack a password of 64 bits. Since the running time is $\Theta(2^n)$ about how days would it take for the program to brute force attack a password of 128 bits?

$$\frac{2^{128}}{2^{64}} \times 9 = 2^{64} \times 9 \text{ days}$$

b) A program requires 9 days to brute force attack a password of 64 bits. About how days would it take for the program to brute force attack a password of 128 bits if the running were $O(n^2)$ instead of exponential?

$$\frac{128^2}{64^2} \times 9 = \left(\frac{128}{64}\right)^2 \times 9 = 4 \times 9 = 36 \text{ days.}$$

15. [6 pts] A particular algorithm on a computer requires 3 seconds to process 50 items and is $\Theta(n^2)$. You want to process 4000 items. You have a choice to either use a computer that is 10 times faster (allowing it to process 50 items in 0.3 seconds) or use the same computer with a different algorithm that still processes 50 items in 3 seconds, but has a growth rate that is $\Theta(n)$.

n t
50 3

sol:

a) Which is the faster choice for 4000 items?

Faster computer: $\frac{4000^2}{50^2} \times 0.3 = 6400 \times 0.3 = 1920$ seconds.

different algorithm: $\left(\frac{4000}{50}\right) \times 3 = 240$ seconds

∴ The different algorithm is the faster choice for 4000 items.

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

sol: suppose input is n

$$\frac{n^2}{50^2} \times 0.3 < \frac{n}{50} \times 3 \qquad \frac{n}{50} \times 0.1 < 1 \Rightarrow n < 500 \text{ items.}$$

∴ when input sizes smaller than 500 items, the faster computer better.

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

① On the same computer, compare with $\Theta(n)$

$$\frac{n^2}{50^2} \times 3 < \frac{n}{50} \times 3$$

$$\frac{n}{50} < 1$$

$n < 50$ items

When $n < 50$ items, $\Theta(n^2)$ better

② On the faster computer

same as b) $n < 500$ items.

③ if different algorithms both on faster computer.

$$\frac{n^2}{50^2} \times 0.3 < \frac{n}{50} \times 0.3$$

$$\frac{n}{50} < 1 \quad n < 50 \text{ items.}$$

④ If the question wrong. mean $\Theta(n)$. Then $n > 500$ items.