CSE 7350 – Test 2
November 2, 2022

Name: _____

- 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 13 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:_____

DATE:_____

CSE 5/7350 – Test #2
November 2, 2022

Name: _____

ID: _____

1. [9 pts] Consider heaps stored in an array:
   (i) How many swaps (maximum) may be required to insert an element into a heap stored as an array that currently has 3 integers?

   2

   (ii) How many swaps (maximum) may be required to delete an element into a heap stored as an array that currently has 9 integers?

   3 (4 if you count the first swap to the top)

   (iii) How many swaps (maximum) may be required to create a heap from an array of 15 integers?

   11

2. [6 pts] If a smallest last ordering has the largest degree when deleted of 13 and a terminal clique size of 11

   (i) What is the maximum number of colors that might be required by the ordering?

   14

   (ii) What is the minimum number of colors that must be required by the graph?

   11

3. [8 pts] When computing n Choose r (nCr), we can use the recursive equation of

$$nCr = (n-1)C(r) + (n-1)C(r-1)$$

Note that $nC0 = 1$ and $nCn = 1$

(i) Show pseudocode of how you would implement a nieve recursive function to compute nCr.

```
Choose (n, r) {
    if (n = 0, r = 0 or n = r) return 1
    return choose(n-1, r) + choose(n-1, r-1)
}
```

(ii) What is the approximate asymptotic bound of the function representing the running time of your code?

$\sim O(2^n)$

(iii) Add a table to your recursive function to improve the running time.

T[n, r] set T[0, all] = 1 T[all, 0] = 1

set T[rest] = -1

```
choose(n, r) {
    if T[n, r] == -1
        T[n, r] = choose(n-1, r) + choose(n-1, r-1)
    return T[n, r]
```

(iv) What is the new asymptotic bound of the function representing the running time of your code?

$\Theta(r \cdot n)$

4. [10 pts] Set up the table as shown in class for the Extended Euclidian Algorithm and compute 1/31 mod 12597

	A	B	q	R	α	β
-1					1	0
0	12597	31	406	11	0	1
1	31	11	2	9	1	-406
2	11	9	1	2	-2	813
3	9	2	4	1	3	-1219
4	2	1	2	0	-14	5689

$\frac{1}{31} \equiv 12597 - 5689 \equiv 5684$

5. [5 pts] Show the swaps required to make a MIN heap using the HEAPIFY algorithm from the following array. Use one swap for each row in the table. Add extra rows if needed.

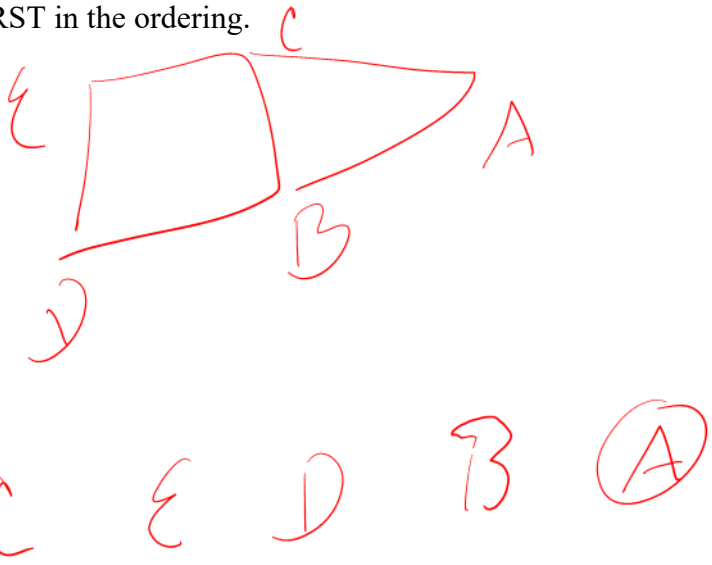
Test #3 for Spr 2023

Index:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Value:	**23**	**6**	**12**	**62**	**89**	**10**	**60**	**33**	**45**	**47**	**21**	**19**	**13**	**85**	**61**	**20**	**30**	**41**
									41									45
								20								33		
							13						60					
					41				89									
				20				62										
								36									62	
			10			12												
	6	23																

20 23

6. [8 pts] Setup the table to find the longest increasing sub-sequence of the following sequence: 2 5 9 6 1 7 4 8

7. [6 pts] Draw a graph and give a smallest last vertex ordering of that graph where the terminal clique is not the largest complete subgraph. Circle the vertex you wrote down FIRST in the ordering.



8. [8 pts] Set up a table to compute the length of the Longest Common Subsequence for the following two strings:

A C T T C G C C and C T A C G A C

		A	C	T	T	C	G	C	C
	0	0	0	0	0	0	0	0	0
C	0	0	1	1	1	1	1	1	1
T	0	0	1	2	2	2	2	2	2
A	0	1	1	2	2	2	2	2	2
C	0	1	2	2	2	3	3	3	3
G	0	1	2	2	2	3	4	4	4
A	0	1	2	2	2	3	4	4	4
C	0	1	2	2	2	3	4	5	5

9. [8 pts] Set up a table to compute the length of the Levenshtein Edit Distance for the following two strings:

A C T T C G C C and C T A C G A C

		A	C	T	T	C	G	C	C
	0	1	2	3	4	5	6	7	8
C	1	1	1	2	3	4	5	6	7
T	2	2	2	1	2	3	4	5	6
A	3	2	3	2	2	3	4	5	6
C	4	3	2	3	3	2	3	4	5
G	5	4	3	3	4	3	2	3	4
A	6	5	4	4	4	4	3	3	4
C	7	6	5	5	5	4	4	3	3

LZW DECODE:

```
read a character k
entry = dictionary entry for k
output entry
w = entry
loop
   read a character k
   entry = dictionary entry for k
   output entry
   add w + first char of entry to the dictionary
   w = entry
endloop
```

10. [8 pts] You have received a message that was compressed with LZW. Remember that A=65, B=66, C=67, and D=68. The dynamic part of the dictionary starts with entry 256. The message you received was

66 65 66 68 257 259 260

(i) What was the original message and what is your dictionary after decompression?

w	k	entry	output	add new
—	66	B	B	
B	65	A	A	BA
A	66	B	B	AB
B	68	D	D	BD
D	257	AB	AB	DA
AB	259	DA	DA	ABD
DA	260	ABD	ABD	DAA

256 = BA
257 = AB
258 = BD
259 = DA
260 = ABD

(ii) Assuming 8 bits per character, how many bits were in the uncompressed message?

11 · 8 = 88

(iii) Assuming the last entry of your dictionary was 2047, how many bits were in the compressed message

7 entries × 11 bits each = 77 bits

(iv) Why might a larger dictionary increase the size of the compressed file?

more bits per entry in the file

11. [8 pts]The Levensthein Edit Distance determines the edit distance between two strings when Addition, Deletion and Substitution are allowed all at a cost of 1.

Assume you have two strings: A and B. the i^{th} character of A is Ai and the j^{th} character of B is Bj.

a. When considering the i^{th} character of A and the j^{th} character of B, what is the formula you would use for determining the value placed in the table at location i,j?

$$T[i,j] = \min \begin{cases} T[i-1,j]+1 \\ T[i,j-1]+1 \\ T[i-1,j-1] + \begin{cases} 0 \text{ when } A[i] = B[j] \\ 1 \text{ when } A[i] \neq B[j] \end{cases} \end{cases}$$

You have been given a new, string processing system that requires 1 cycle to delete a character, 2 cycles to substitute a character and 1 cycle to add a character.

b. When converting from string A to string B and considering the i^{th} character of A and the j^{th} character of B, what is the formula you would use for determining the value placed in the table at location i,j?

$$T[i,j] = \min \begin{cases} T[i-1,j]+1 \\ T[i,j-1]+1 \\ T[i-1,j-1] + \begin{cases} 0 \text{ when } A[i] = B[j] \\ 2 \text{ when } A[i] \neq B[j] \end{cases} \end{cases}$$

c. Fill in the table to determine the minimum number of cycles required to convert from string A = S G P Z T to string B = T S Z T M

		S	G	P	Z	T	
	0	1	2	3	4	5	
T	1	2	3	4	5	4	
S	2	1	2	3	4	5	
Z	3	2	3	4	3	4	
T	4	3	4	5	4	3	
M	5	4	5	6	5	4	

d. Using your table above, what is the minimum number of cycles required to convert from string A = S G P Z T to string B = T S Z T M

4

12. [8 pts] You have 3 different dice. Dice 1 has sides {1,2,3}. Dice 2 has sides {2,2,2,3,3,3,4,4,4} and Dice 3 has sides {2,3,3,4}. How many ways can you roll a 9 with these three dice? Set up the table for the dynamic programming algorithm and fill in the complete columns for Dice 1 and Dice 2. You may only fill in as much as you wish for Dice 3.

0	0	0	0
1	1	0	0
2	1	0	0
3	1	3	0
4	0	6	0
5	0	9	3
6	0	6	15
7	0	3	24
8	0	0	30
9	0	0	24

13. [8 pts] You have 2 different dice that are not evenly weighted:
    - Dice 1 has sides {1,2,3} and a 10% chance of rolling a 1, a 40% chance of rolling a 2 and a 50% chance of rolling a 3.
    - Dice 2 has sides {2,2,3,3,4,4} with a 15% chance for each 2, a 15% chance for each 3 and a 20% chance for each 4.
    - What is the probability of rolling a 6 with these dice? Set up the table for the dynamic programming algorithm and fill in the complete column for Dice 1 and Dice 2.

31%

0	0	0
1	10%	0
2	40%	0
3	50%	3%
4	0	15%
5	0	31%
6	0	31%
7	0	20%

Scratch Paper

Scratch Paper