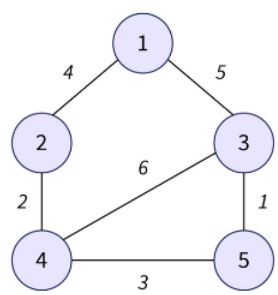
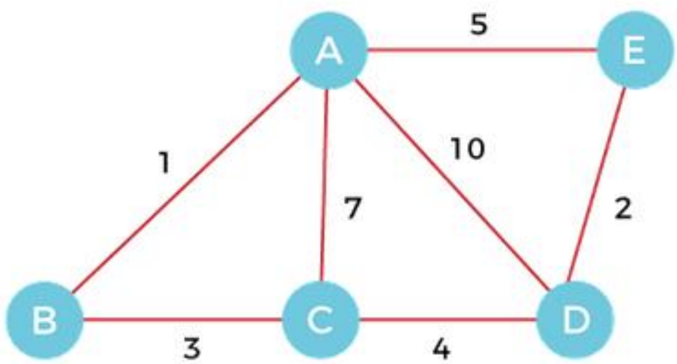
Department of Computer Science and Engineering
M.Tech in Computer Science and Engineering (CSE)
Continuous Internal Evaluation (CIE-III)- Question Paper

Course	Advanced Data Structures and Algorithms	Course Code: 22MCE12TL (MCE201I)	Sem: 01
15.05.2024	Duration : 90 minutes	Max Marks: 50	Staff: RS

Sl. No.	Answer all questions	M	* L1-L6	**CO
1a.	Apply Prim's algorithm for the Graph given below and find the minimum spanning tree and its cost. 	7	L3	CO1
1b.	Differentiate between the working of Naïve and Rabin Karp algorithm.	3	L3	CO1
2a.	Apply Rabin Karp algorithm and search for the Pattern in the Text. Discuss the time complexity of the algorithm. Text: 7896234567897896 Pattern:896	7	L4	CO2
2b.	Construct a graphical representation of a trie that contains the following data with the key being built as you descend down from the root to a leaf. For example, the word, "ape" is built by traversing the "a" edge, the "p" edge, and the "e" edge. When leaf node is reached, the value 32 is stored at the leaf node. map = { 'ape': 32, 'ball': 2, 'atom': 16, 'ate': 18, 'bait': 5 }	3	L3	CO2
3a.	Let's say we have a trie that has the following words in it already. **home, house, belated, heated** If we add the following words, how many nodes will be added to the trie? Show the graphical representation of the trie tree. **hose, belt, heal**	8	L4	CO3
3b.	List the major applications of Trie.	2	L2	CO4

4a.	Apply Kruskal's algorithm for the Graph given below and find the minimum spanning tree and its cost.  	6	L4	CO4
4b.	Comapre Trie datastructure with Hash tables.	4	L4	CO3
5a.	Consider a Hash Table with 9 slots. Apply suitable hash function and insert the the following keys in the order: 5, 28, 19, 15, 20, 33, 12, 17, 10. Resolve the collisions using chaining technique. Also mention the maximum, minimum and average chain lengths in the Hash tables, respectively.	6	L4	CO2
5b.	Consider a list of numbers 34,16, 2, 93, 80, 77, 51 with Hash table size 10. Mention the order of the elements in the Hash table from index 0 to size-1 after applying the hash function (Key%HashtableSize).	4	L3	CO3

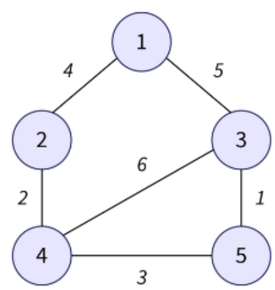
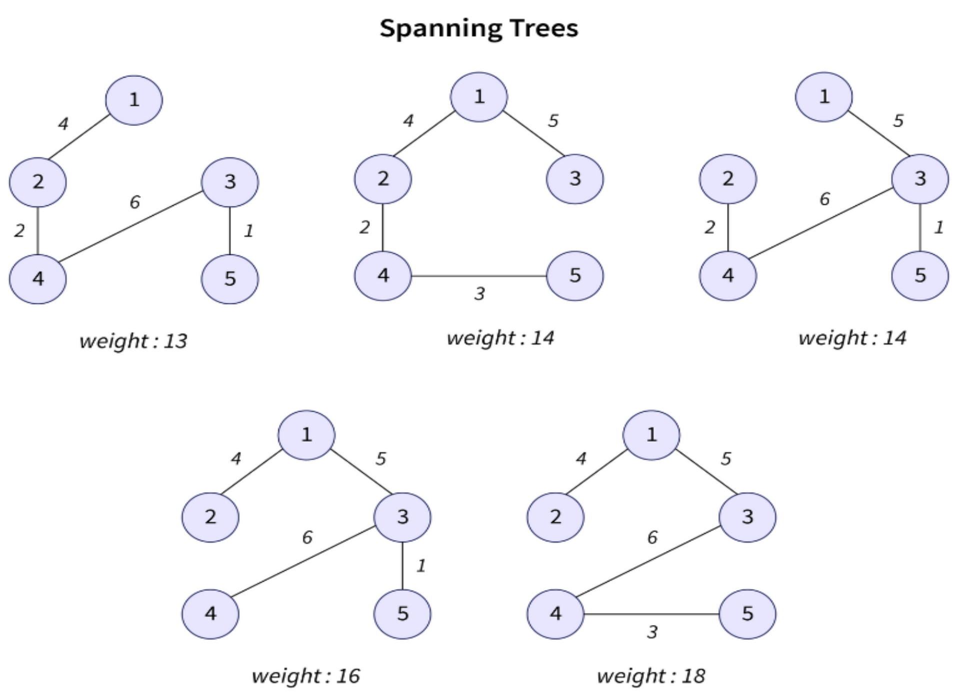
****Course Outcome** CO1: Analyze the efficiency of programs based on time complexity. CO2: Critically think and apply appropriate design paradigm and algorithm for a specific problem. CO3: Apply knowledge of computing and mathematics to algorithm design. CO4: Design, implement and evaluate algorithms to solve real world problems

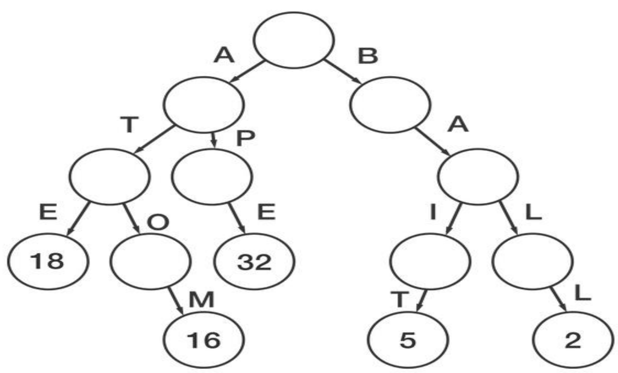
Marks Distribution *(L1-L6)

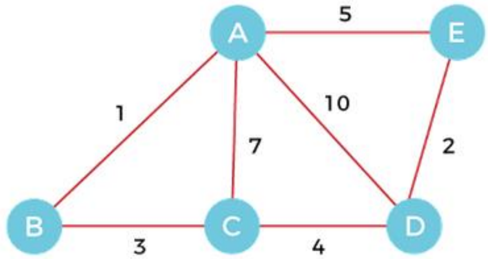
L1	L2	L3	L4	L5	L6	CO1	CO2	CO3	CO4
0	2	17	31	0	0	10	16	16	8

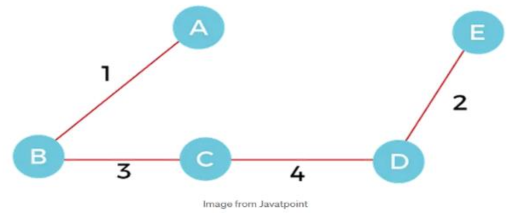
Department of Computer Science and Engineering
M.Tech in Computer Science and Engineering (CSE)
Continuous Internal Evaluation (CIE-III)-Scheme and Solution

Course	Advanced Data Structures and Algorithms	Course Code: 22MCE12TL (MCE201I)	Sem: 01
15.05.2024	Duration : 90 minutes	Max Marks: 50	Staff: RS

Sl. No.	**Answer all questions**	**M**
1a.	Apply Prim's algorithm for the Graph given below and find the minimum spanning tree and its cost.  	7
1b.	Like the Naive Algorithm, the Rabin-Karp algorithm also check every substring. But unlike the Naive algorithm, the Rabin Karp algorithm matches the hash value of the pattern with the hash value of the current substring of text, and if the hash values match then only it starts matching individual characters	3

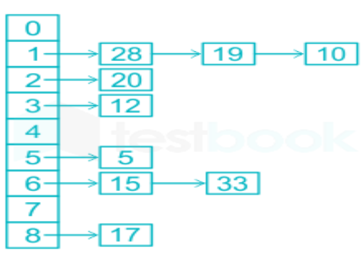
2a.	Apply Rabin Karp algorithm and search for the Pattern in the Text. Discuss the time complexity of the algorithm. Text: 7896234567897896 Pattern:896	7
2b.	Construct a graphical representation of a trie that contains the following data with the key being built as you descend down from the root to a leaf. For example, the word, "ape" is built by traversing the "a" edge, the "p" edge, and the "e" edge. When leaf node is reached, the value 32 is stored at the leaf node. map = { 'ape': 32, 'ball': 2, 'atom': 16, 'ate': 18, 'bait': 5 }  A trie	3
3a.	Let's say we have a trie that has the following words in it already. **home, house, belated, heated** If we add the following words, how many nodes will be added to the trie? Show the graphical representation of the trie tree. **hose, belt, heal** **4 nodes will be added to the Trie**	8
3b.	Applications of Trie • Sorting. Lexicographic sorting of a set of string keys can be implemented by building a trie for the given keys and traversing the tree in pre-order fashion; this is also a form of radix sort. ... • Full-text search • Web search engines • Bioinformatics • Internet routing	2

<table>
<tr><td>4a.</td><td>Apply Kruskal’s algorithm for the Graph given below and find the minimum spanning tree and its cost.




The cost of the MST is = AB + DE + BC + CD = 1 + 2 + 3 + 4 = 10.</td><td>6</td></tr>
<tr><td>4b.</td><td>Comapre Trie datastructure with Hash tables

Predictable O(n) lookup time where n is the size of the key.
Lookup can take less than n time if it's not there.
Supports ordered traversal.
No need for a hash function.
Deletion is straightforward.
You can quickly look up prefixes of keys, enumerate all entries with a given prefix
</td><td>4</td></tr>
<tr><td>5a.</td><td>Consider a Hash Table with 9 slots. Apply suitable hash function and insert the the following keys in the order: 5, 28, 19, 15, 20, 33, 12, 17, 10. Resolve the collisions using chaining technique. Also mention the maximum, minimum and average chain lengths in the Hash tables, respectively.

Keys: 5, 28, 19, 15, 20, 33, 12, 17,
10.
h(k) = k mod 9


Chaining
Maximum chain length = 3 (28 -> 19 -> 10)
Minimum chain length = 0 (0, 4, 7 slot doesn’t have any element)
$\textbf{Average chain length} = \frac{0+3+1+1+1+0+1+2+0+1}{9} = \frac{9}{9} = 1$</td><td>6</td></tr>
<tr><td>5b.</td><td>Consider a list of numbers 34,16, 2, 93, 80, 77, 51 with Hash table size 10. Mention the order of the elements in the Hash table from index 0 to size-1 after applying the hash function (Key%HashtableSize).</td><td>4</td></tr>
</table>

The given data,

list of numbers (34, 16, 2, 93, 80, 77, 51) and has a table size is 10.

Element	Hash function	Index
34	34%10 =4	4
16	16%10 =6	6
2	2%10 =2	2
93	93 % 10 = 3	3
80	80 %10 =0	0
77	77% 10= 7	7
51	51 % 10 = 1	1

The hash table is,

index	Value
0	80
1	51
2	2
3	93
4	34
5	
6	16
7	77
8	
9	

The remaining index stores the null values.

Hence the correct answer is 80, 51, 2, 93, 34, null, 16, 77, null, null.