TERAFAC TECHNOLOGIES PRIVATE LIMITED

Plot no. 57, Industrial Area Phase I, Chandigarh, 160002
website: www.terafac.com

TERAFAC TECHNOLOGIES Pvt. Ltd.

Technical Test Time Limit: 60 mins

Easy:

1. Write a short Python function, is multiple(n, m), that takes two integer values and returns True if n is a multiple of m, that is, n = mi for some integer i, and False otherwise.
2. Write a short Python function that counts the number of vowels in a given character string.
3. Write a short Python program that takes two arrays a and b of length n storing int values, and returns the dot product of a and b. That is, it returns an array c of length n such that c[i] = a[i] · b[i], for i = 0,...,n−1.
4. What values are returned during the following series of stack operations, if executed upon an initially empty stack? push(5), push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop(), push(7), push(6), pop(), pop(), push(4), pop(), pop().

Medium:

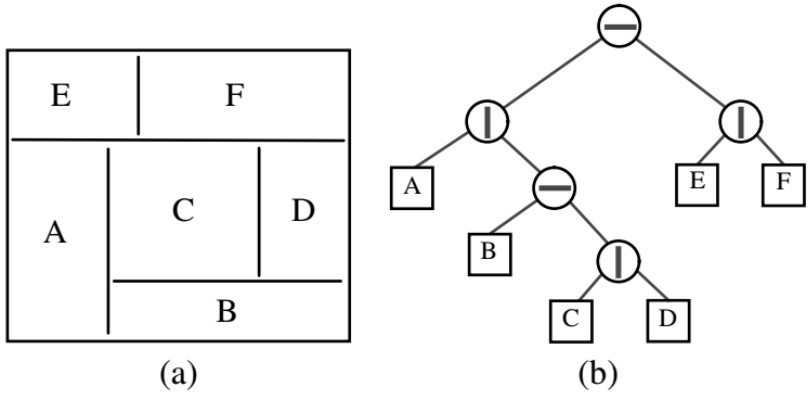
1. Write a Python program that inputs a polynomial in standard algebraic notation and outputs the first derivative of that polynomial.
2. Write a set of Python classes that can simulate an Internet application in which one party, Alice, is periodically creating a set of packets that she wants to send to Bob. An Internet process is continually checking if Alice has any packets to send, and if so, it delivers them to Bob's computer, and Bob is periodically checking if his computer has a packet from Alice, and, if so, he reads and deletes it.
3. A sequence S contains n − 1 unique integers in the range [0,n − 1], that is, there is one number from this range that is not in S. Design an O(n)-time algorithm for finding that number. You are only allowed to use O(1) additional space besides the sequence S itself.
4. Give a recursive algorithm to compute the product of two positive integers,m and n, using only addition and subtraction.

Hard:

1. Suppose Alice has picked three distinct integers and placed them into a stack S in random order. Write a short, straight-line piece of pseudo-code (with no loops or recursion) that uses only one comparison and only one variable x, yet that results in variable x storing the largest of Alice's three integers with probability 2/3. Argue why your method is correct.
2. Draw a binary tree T that simultaneously satisfies the following: • Each internal node of T stores a single character. • A preorder traversal of T yields EXAMFUN. • An inorder traversal of T yields MAFXUEN.

Technical Test Time Limit: 60 mins

3. A slicing floor plan divides a rectangle with horizontal and vertical sides using horizontal and vertical cuts. (See Figure 8.25a.) A slicing floor plan can be represented by a proper binary tree, called a slicing tree, whose internal nodes represent the cuts, and whose external nodes represent the basic rectangles into which the floor plan is decomposed by the cuts. The compaction problem for a slicing floor plan is defined as follows. Assume that each basic rectangle of a slicing floor plan is assigned a minimum width w and a minimum height h. The compaction problem is to find the smallest possible height and width for each rectangle of the slicing floor plan that is compatible with the minimum dimensions



of the basic rectangles. Namely, this problem requires the assignment of values h(p) and w(p) to each position p of the slicing tree such that:

TERAFAC TECHNOLOGIES PRIVATE LIMITED

Plot no. 57, Industrial Area Phase I, Chandigarh, 160002

website: www.terafac.com

Technical Test

Time Limit: 60 mins

$$\mathsf{w}(p) = \begin{cases} w & \text{if } p \text{ is a leaf whose basic rectangle has minimum width } w \\ \max(\mathsf{w}(\ell), \mathsf{w}(r)) & \text{if } p \text{ is an internal position, associated with a horizontal cut, with left child } \ell \text{ and right child } r \\ \mathsf{w}(\ell) + \mathsf{w}(r) & \text{if } p \text{ is an internal position, associated with a vertical cut, with left child } \ell \text{ and right child } r \end{cases}$$

$$\mathsf{h}(p) = \begin{cases} h & \text{if } p \text{ is a leaf node whose basic rectangle has minimum height } h \\ \mathsf{h}(\ell) + \mathsf{h}(r) & \text{if } p \text{ is an internal position, associated with a horizontal cut, with left child } \ell \text{ and right child } r \\ \max(\mathsf{h}(\ell), \mathsf{h}(r)) & \text{if } p \text{ is an internal position, associated with a vertical cut, with left child } \ell \text{ and right child } r \end{cases}$$

Design a data structure for slicing floor plans that supports the operations:

- Create a floor plan consisting of a single basic rectangle.
- Decompose a basic rectangle by means of a horizontal cut.
- Decompose a basic rectangle by means of a vertical cut.
- Assign minimum height and width to a basic rectangle.
- Draw the slicing tree associated with the floor plan.
- Compact and draw the floor plan.