YOU WILL

> CRACK ACCENTURE

ON & OFF CAMPUS

CODING QUE SHEET

Accenture coding round Cheatsheet

- **Reverse a String**: Write a function that reverses a given string.

- **Find the Maximum Element in an Array**: Write a function that finds the maximum element in a given array of integers.

- **Check if a String is a Palindrome**: Write a function that checks if a given string reads the same backward as forward.

- **Find the Missing Number in an Array**: Given an array of consecutive integers with one missing element, write a function to find the missing number.

- **Find the Intersection of Two Arrays**: Write a function that finds the elements present in both a given pair of arrays.

- **Remove Duplicates from an Array**: Write a function that removes duplicate elements from a given array while preserving the order.

- **Count the Occurrence of Each Element in an Array**: Write a function that counts the number of times each element appears in a given array.

- **Find the Second Largest Element in an Array**: Write a function that finds the second largest element in a given array.

- **Merge Two Sorted Arrays**: Write a function that merges two sorted arrays into a single sorted array.

- **Find the Length of the Longest Substring Without Repeating Characters**: Write a function that finds the length of the longest substring in a string where all characters are unique.

- **Reverse a Linked List**: Write a function that reverses a given linked list.

- **Implement a Stack Using an Array**: Write code to implement a stack data structure using an array.

- **Implement a Queue Using an Array**: Write code to implement a queue data structure using an array.

- **Find the Lowest Common Ancestor in a Binary Tree**: Write a function to find the lowest common ancestor (LCA) of two given nodes in a binary tree.

- **Check if a Binary Tree is a Binary Search Tree (BST)**: Write a function that checks if a given binary tree adheres to the properties of a BST.

- **Find the Level Order Traversal of a Binary Tree**: Write a function that performs a level-order traversal of a binary tree, returning the elements level by level.

- **Clone a Linked List with Random Pointers**: Write a function that creates a deep copy of a linked list where each node also has a random pointer to another node in the list.

- **Find the Kth Largest Element in an Array**: Write a function that finds the kth largest element in a given array.

- **Find All Combinations of a Sum**: Write a function that finds all unique combinations of numbers in a given array that add up to a specific target sum.

- **Binary Operations on Strings**: Write a function that performs binary operations like AND, OR, and XOR on a given string of binary numbers with respective operators.

- **Calculate Binary Operations**: Given a string with binary numbers and operators (AND, OR, XOR), write a function to calculate the result by scanning from left to right.

- **Find the Maximum and Its Index in an Array**: Write a function to find the greatest number in an array and its index.

- **Operation Choices**: Write a function that performs operations (addition, subtraction, multiplication, division) based on a given integer parameter.

- **Food Distribution Among Rats**: Given the number of rats and their food requirement, write a function to determine if the food available in different houses is sufficient.

- **You are given a function, intMaxExponents(int a, int b) Find and return the number between 'a' and 'b'(range inclusive on both ends) which has the**

maximum exponent of 2.

- **Execute this function of Accenture Coding Questions: def differenceofSum(n.m)**

- **Form an array of 1000 integers and find out the second-largest number. If there is no second-largest number, return the value to −1.**

- **Adam decides to do some charity work. From day 1 till day x, he will give i^2 coins to charity. On the day 'i' (1 < = i < = x), find the number of coins he gives to charity.**

- **Find the sum of the divisors for the N integer number.**

- **Write a program that accepts the integer array of length 'size' and finds the largest number that can be formed by permutation.**

- **Write a function SmallLargeSum(array) which accepts the array as an argument or parameter, that performs the addition of the second largest element from the even location with the second largest element from an odd location?**

  - **Rules**

  a. All the array elements are unique.

  b. If the length of the array is 3 or less than 3, then return 0.

  c. If Array is empty then return zero.

- **Write a function CheckPassword(str) which will accept the string as an argument or parameter and validates the password. It will return 1 if the conditions are satisfied else it'll return 0?**

  - **The password is valid if it satisfies the below conditions:**

  - **a. It should contain at least 4 characters.**

  - **b. At least 1 numeric digit should be present.**

  - **c. At least 1 Capital letter should be there.**

  - **d. Passwords should not contain space or slash(/).**

  - **e. The starting character should not be a number.**

PRIME CODING

- **Write a function FindMaxInArray, which will find the greatest number from an array with its desired index? The greatest number and its desired index should be printed in separate lines.**

- **Write a function OperationChoices(c, a, b) which will accept three integers as an argument, and the function will return:**
  - **a. (a + b) if the value of c=1.**

    b. (a – b) if the value of c=2.

    c. (a b) if the value of c=3.

    d. (a / b) if the value of c=4.

- **You are required to input the size of the matrix then the elements of matrix, then you have to divide the main matrix in two sub matrices (even and odd) in such a way that element at 0 index will be considered as even and element at 1st index will be considered as odd and so on. Then you have sort the even and odd matrices in ascending order and print the sum of second largest number from both the matrices.**

- **Execute the given function.**
  - def differenceofSum(n.m)

    The function takes two integrals m and n as arguments. You are required to obtain the total of all integers ranging between 1 to n (both inclusive) which are not divisible by m. You must also return the distinction between the sum of integers not divisible by m with the sum of integers divisible by m.

    Assumption

    m > 0 and n > 0

    Sum lies within the integral range

- **Execute the given function.**
  - def LargeSmallSum(arr)

    The function takes an integral arr which is of the size or length of its arguments. Return the sum of the second smallest element at odd position 'arr' and the second largest element at the even position.

Assumption

- Every array element is unique.

- Array is 0 indexed.

- **Write a function to validate if the provided two strings are anagrams or not. If the two strings are anagrams, then return 'yes'. Otherwise, return 'no'.**

- **The given function has a string (str) and two characters, ch1 and ch2. Execute the function in such a way that str returns to its original string, and all the events in ch1 are replaced by ch2, and vice versa.**

    - Replacecharacter(Char str1, Char ch1, Int 1, Char ch2)

- **Write a function to find roots of a quadratic equation $ax^2 + bx + c = 0$.**

- **Find the sum of the divisors for the N integer number.**

- **Perform a function to reverse a string word-wise. The input here will be the string. In the output, the last word mentioned should come as the first word and vice versa**

- **Write a program to count the number of swaps required to sort a given list of integers in ascending order using the selection sort algorithm.**

- The function accepts an integer array 'arr' of size 'n' as its argument. Each element of 'arr' represents the number of chocolates distributed to a person. The function needs to return the minimum number of chocolates that need to be distributed to each person so that the difference between the chocolates of any two people is minimized.

- The function accepts a character array 'arr' of size 'n' as its argument. Each element of 'arr' represents the status of a parking slot, where 'S' represents an empty slot and 'X' represents an occupied slot. The function needs to return the maximum number of cars that can be parked in the parking lot. It is assumed that two cars cannot occupy the same slot and cars can only park in consecutive empty slots.

- The function accepts a string 'str' as its argument. The function needs to return the transformed string by replacing all occurrences of the character 'a' with the character 'b' and vice versa.

- The function accepts an integer array 'arr' of size 'n' as its argument. The function needs to return the index of an equilibrium point in the array, where the sum of elements on the left of the index is equal to the sum of elements on the right of the index. If no equilibrium point exists, the function should return -1.

- The function accepts an integer array 'arr' of size 'n' and an integer 'd' as its argument. The function needs to rotate the array 'arr' by 'd' positions to the right. The rotation should be done in place, without using any additional memory.

- The function accepts two strings 'str1' and 'str2' as its argument. The function needs to return the index of the first occurrence of substring 'str2' in string 'str1' or -1 if the substring is not found.

- The function accepts a string 'str' as its argument. The function needs to reverse the order of the words in the string.

- Given an array of integers and an integer sum, find a pair of numbers (a, b) in the array where a + b = sum.

- Given an array of integers, find the maximum subarray sum. A subarray is a contiguous subsequence of the array.

- Given a string str, a character ch1, and a character ch2, replace all occurrences of ch1 in str with ch2 and vice versa.

- Given an integer array, find the minimum value and its index in the array.

- Given an array of integers, find the average of all positive numbers in the array.

- Given an integer array and an integer element, count the number of occurrences of the element in the array.

- Given an integer array and an integer element, check if the array contains the element.