Siliguri Institute of Technology
Master of Computer Application
Object Oriented Programming with JAVA
Subject Code: MCAN-293

Assignment II [Array & String]

Date of Submission: 20th March, 2025

Q1. Longest Common Prefix

Write a function to find the longest common prefix string amongst an array of strings. If there is no common prefix, return an empty string "".

Example 1:

Input: strs = ["flower","flow","flight"]

Output: "fl"

Example 2:

Input: strs = ["dog","race","car"]

Output: ""

Explanation: There is no common prefix among the input strings.

Q2. Password Checker

You are given a function.

int CheckPassword(char str[], int n);

The function accepts string str of size n as an argument. Implement the function which returns 1 if given string str is valid password else 0.

str is a valid password if it satisfies the below conditions.

- At least 4 characters
- At least one numeric digit
- At Least one Capital Letter
- Must not have space or slash (/)
- Starting character must not be a number

Assumption: Input string will not be empty.

Example:

Input:

aA1_67

Output:

1

Sample Input:

a987 abC012

Output:

0

Q3. Letter Combinations of a Phone Number

Given a string containing digits from 2-9 inclusive, return all possible letter combinations that the number could represent. Return the answer in any order.

A mapping of digit to letters (just like on the telephone buttons) is given below. Note that 1 does not map to any letters.

Example 1:
Input: digits = "23"
Output: ["ad","ae","af","bd","be","bf","cd","ce","cf"]
Example 2:
Input: digits = ""
Output: []
Example 3:
Input: digits = "2"
Output: ["a","b","c"]

Q4. Java program to find all permutations of a given String using recursion.
For example, given a String "XYZ", this program will print all 6 possible permutations
e.g. XYZ, XZY, YXZ, YZX, ZXY, XYX

Q5. Anagram Check
Given two strings s and t, return true if t is an anagram of s, and false otherwise.
Example 1:
Input: s = "anagram", t = "nagaram"
Output: true
Example 2:
Input: s = "rat", t = "car"
Output: false
Here are some **case study-based programming questions** on **Java Arrays and Strings**, along with **sample input and output**.

Q6. Replace Profanity in Text

Problem Statement:

A social media platform wants to automatically replace **offensive words** with `***` in user-generated content. Given a **list of bad words**, replace them in the input text.

Example Input/Output:

Bad Words List: `"bad", "ugly", "stupid"`

Input:

```
Enter text: This is a bad and ugly comment.
```

Output:

```
Filtered text: This is a *** and *** comment.
```

Q7. Unique Characters in a String

Problem Statement:
A company is developing a new security system that requires checking whether a given string has all unique characters. You need to write a program that:

- Accepts a string from the user.
- Checks if all characters in the string are unique.

Example Input/Output:
Input 1:
Enter a string: abcde
Output 1:
All characters are unique.
Input 2:
Enter a string: hello
Output 2:
String contains duplicate characters.

Q8. Find the Longest Word

Problem Statement:

A book editor tool requires a function to find the **longest word** in a given sentence.

Example Input/Output:

Input:

```
Enter sentence: Innovation distinguishes between a leader and a follower.
```

Output:

```
Longest Word: distinguishes
```

Q9. Convert Sentence to Title Case

Problem Statement:

A content editing software needs a function to convert a **sentence** into **title case** (i.e., the first letter of each word should be uppercase).

Example Input/Output:

Input:

```
Enter sentence: java programming is fun
```

Output:

```
Title Case: Java Programming Is Fun
```

Q10. Extract Domain from URL

Problem Statement:

A digital marketing company wants to extract the **domain name** from a given URL. Write a Java program that:

- Accepts a **URL**.
- Extracts and displays only the **domain name**.

Example Input/Output:

Input:

```
Enter URL: https://www.google.com/search?q=java
```

Output:

```
Domain: google.com
```

***Note: You are supposed to use at least two classes and one object for each program.