Nibila Amutha

1. Given a string s, return *the longest palindromic substring* in s. A string is called a palindrome string if the reverse of that string is the same as the original string.

Input: s = "babad"
Output: "bab"
Explanation: "aba" is also a valid answer.

Input: s = "cbbd"
Output: "bb"

2. 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 "".

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

Input: strs = ["dog","racecar","car"]
Output: ""
Explanation: There is no common prefix among the input strings.

3. Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid. An input string is valid if:

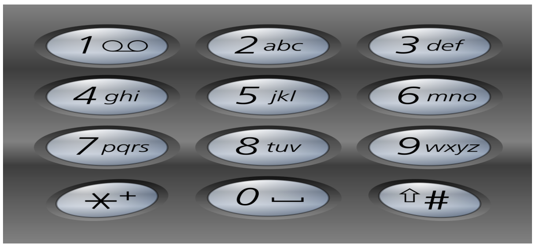
1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.
3. Every close bracket has a corresponding open bracket of the same type.

Input: s = "()"
Output: true

Input: s = "()[]{}"
Output: true

Input: s = "(]"
Output: false

4. 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**.cA mapping of digits to letters (just like on the telephone buttons) is given below. Note that 1 does not map to any letters.



Input: digits = "23"
Output: ["ad","ae","af","bd","be","bf","cd","ce","cf"]

Input: digits = ""
Output: []

Input: digits = "2"
Output: ["a","b","c"]

5. Given an integer n, return *a string array* answer ***(1-indexed)*** *where*:

- answer[i] == "FizzBuzz" if i is divisible by 3 and 5.
- answer[i] == "Fizz" if i is divisible by 3.
- answer[i] == "Buzz" if i is divisible by 5.
- answer[i] == i (as a string) if none of the above conditions are true.

Input: n = 3
Output: ["1","2","Fizz"]

Input: n = 5
Output: ["1","2","Fizz","4","Buzz"]

Input: n = 15
Output:
["1","2","Fizz","4","Buzz","Fizz","7","8","Fizz","Buzz","11","Fizz","13","14","FizzBuzz"]

6.Given two binary strings a and b, return their sum as a binary string.

Input: a = "11", b = "1"
Output: "100"

Input: a = "1010", b = "1011"
Output: "10101"