Problem Statement:

With the right choice of a data structure, organize the contact details of various businesses (like Just Dial).Look up could be based on category or a specific business name. Also enable reverse phone lookup.

Introduction

In today's fast-paced world, businesses and customers require a streamlined way to access and organize contact details. With the right choice of data structures, businesses can effectively manage their contact information, making it easier for both business owners and customers to access key details. The idea behind creating a business directory system, similar to services like **Just Dial**, is to organize the contact details of various businesses in such a way that users can search for them based on categories, business names, or even phone numbers. This approach ensures a robust and efficient solution for managing business contacts in real time.

A business directory system typically includes the following features:

- **Business name**: The name of the business.
- **Category**: The type or industry of the business (e.g., restaurant, hospital, etc.).
- **Address**: The location of the business.
- **Phone numbers**: A list of phone numbers that customers can use to contact the business.

Additionally, an essential feature in modern business directories is **reverse phone lookup**. This allows users to search for businesses by phone numbers, enabling quick identification of the business associated with a particular number.

In this report, we will present an algorithm that efficiently manages and stores business contact details, allowing lookups by category, business name, and reverse phone lookup.

Algorithm for the Business Directory System

➢ Step 1 : Initialize Constants and Variables:

- Define constants for maximum entries, name length, category length, address length, phone numbers, and phone number length.
- Initialize arrays:
    - names[MAX_ENTRIES][MAX_NAME_LENGTH] for storing business names.
    - categories[MAX_ENTRIES][MAX_CATEGORY_LENGTH] for storing business categories.
    - addresses[MAX_ENTRIES][MAX_ADDRESS_LENGTH] for storing business addresses.
    - phoneNumbers[MAX_ENTRIES][MAX_PHONE_NUMBERS][MAX_PHONE_LENGTH] for storing phone numbers.
    - phoneCounts[MAX_ENTRIES] for tracking the number of phone numbers associated with each business.

➢ Step 2 : Add Entry (Function: addEntry):

- Check if the directory is full (entryCount >= MAX_ENTRIES).
    - If true, display an error message and return.
- Prompt the user for the business name, category, address, and number of phone numbers.
- Validate and set the number of phone numbers, ensuring it does not exceed the maximum allowed (MAX_PHONE_NUMBERS).
- Input each phone number and store it in the phoneNumbers array.
- Increment entryCount.
    - Display a success message.Print "Entry added successfully!"

➢ Step 3 : Lookup by Category (Function: lookupByCategory):

- Prompt the user for a category to search for.
- For each business entry in the directory, check if the category matches the input.
- If a match is found, display the business details (name, address, and phone numbers).
- If no matches are found, display a message indicating no businesses are found in that category.

➢ Step 4 : Lookup by Business Name (Function: lookupByName):

- Prompt the user for the business name to search for.
- For each business entry, check if the name matches the input.
- If a match is found, display the business details (category, address, and phone numbers).
- If no matches are found, display a message indicating no businesses are found by that name

➢ Step 5 : Reverse Phone Lookup (Function: reversePhoneLookup):

- Prompt the user for a phone number to search for.
- For each business entry, check each of its phone numbers to see if it matches the input.
- If a match is found, display the business details (name, category, and address).
- If no matches are found, display a message indicating no businesses are found with that phone number.

➢ Step 6 : Display Menu (Function: displayMenu):

- Print the main menu with the following options:
  - "1. Add a Business"
  - "2. Lookup by Category"
  - "3. Lookup by Business Name"
  - "4. Reverse Phone Lookup"

- "5. Exit"

- Prompt the user to select a choice.

➢ Step 7 : Main Program Loop:

- Continuously display the menu and prompt the user for a choice.
- Based on the user's input:
    - If choice is 1, call addEntry.
    - If choice is 2, call lookupByCategory.
    - If choice is 3, call lookupByName.
    - If choice is 4, call reversePhoneLookup.
    - If choice is 5, print "Exiting the program. Goodbye!" and terminate the program.
    - For invalid choices, print "Invalid choice. Please try again."
- **End Program** when the user chooses to exit.

Input/Output:

1) Add a Business

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== Business Directory ===
1. Add a Business
2. Lookup by Category
3. Lookup by Business Name
4. Reverse Phone Lookup
5. Exit
Enter your choice: 1
Enter Business Name: Tech Solutions
Enter Category: IT Services
Enter Address: hassan arsikere 573122
Enter number of phone numbers (max 5): 2
Enter Phone Number 1: 9353709621
Enter Phone Number 2: 7337810154
Entry added successfully!
```

2) Lookup by category

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== Business Directory ===
1. Add a Business
2. Lookup by Category
3. Lookup by Business Name
4. Reverse Phone Lookup
5. Exit
Enter your choice: 2
Enter Category to search: IT Services

Businesses in category 'IT Services':
  Name: Tech Solutions
  Address: hassan arsikere 573122
  Phone Numbers:
    9353709621
    7337810154
```

3) Lookup by Business Name:

```
=== Business Directory ===
1. Add a Business
2. Lookup by Category
3. Lookup by Business Name
4. Reverse Phone Lookup
5. Exit
Enter your choice: 3
Enter Business Name to search: Tech Solutions

Business Details:
  Name: Tech Solutions
  Category: IT Services
  Address: hassan arsikere 573122
  Phone Numbers:
    9353709621
    7337810154
```

4) Reverse Phone Lookup

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

=== Business Directory ===
1. Add a Business
2. Lookup by Category
3. Lookup by Business Name
4. Reverse Phone Lookup
5. Exit
Enter your choice: 4
Enter Phone Number to search: 9353709621

Businesses with phone number '9353709621':
  Name: Tech Solutions
  Category: IT Services
  Address: hassan arsikere 573122
```

5) Exit the Program

```
=== Business Directory ===
1. Add a Business
2. Lookup by Category
3. Lookup by Business Name
4. Reverse Phone Lookup
5. Exit
Enter your choice: 5
Exiting the program. Goodbye!
PS C:\Users\namra\OneDrive\Desktop\c c++ program>
```