

PHONEBOOK & SERVICE PROVIDER MANAGEMENT

-SHASHANK SAVARKAR (24CE021)

Objective

- Data Management: To allow users to store, retrieve, update and delete information about specific contact and service provider.
- Functionality: Adding, searching, editing, updating, managing all data in separate text file for each service provider.
- Error handling and Validation: Input validation ensuring phone numbers and handling errors effectively

Basic Code / Structure

- The code contains effective use of file handling, functions, structure etc.
- It contains following functions:

```
void addContact(const char *filePath);  
void viewContacts(const char *filePath);  
void searchContact(const char *filePath);  
void updateContact(const char *filePath);  
void removeContact(const char *filePath);  
void moveContact(const char *sourceFile, const char *targetFile);  
void manageProviderOptions(int providerChoice);
```

Basic Code / Structure

- Here is the structure:

```
typedef struct
```

```
{
```

```
    char name[50];
```

```
    char number[15];
```

```
} Contact;
```

```
typedef struct
```

```
{
```

```
    char name[25];
```

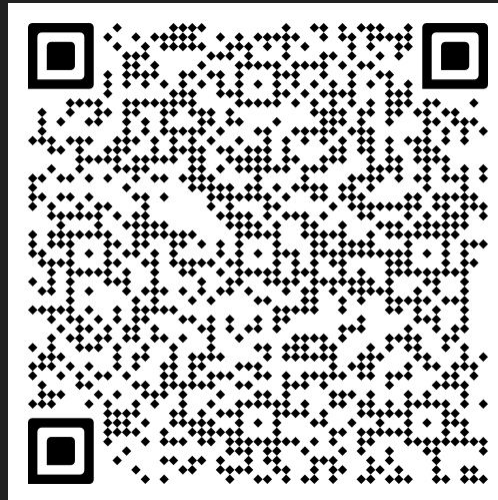
```
    char fileName[25];
```

```
} Company;
```

Source Link

- GitHub Link :

<https://github.com/operashashank/phonebook-serviceprovidermanagementsystem>



Hardships that I overcame

- **File Handling:**Major problem while programming was that i was not understanding the concept of file handling and how to implement it in my code as I was told to create separate text file for the each service operator. Later on I searched through web and went through the notes given by sir.
- **Function to port the number:** Till this moment I am not able to code a function that will change the service provider from one to another keeping the date of previous number in text file.
- **Search Number:** I was unable to code a function that will search a particular contact number. Later on my fellow batchmates taught me about string by which we can create a search algorithm.
- **Improving Program Structure:** I faced lot challenge to reduce the number of lines of codes as 99% of the time reading of code is done by other programmers or faculties so I needed to make sure that the code should be compact and should have less lines of codes.

Problems which lie in the limelight

- How program will behave if it detects invalid number as input from the user?
- What if number is duplicate or a same number is added by different names?
- Port number from one service provider to another service provider.

These are the problems which need to be resolved in the code.

References

- Professor Ramlal Sir - Helped to improve program, provided file handling, structure, function etc notes.
- Programming Pathshala - Helped to improve logical thinking through interactive lectures.
- GeeksforGeeks: Helped to understand syntax.