

# Section A: Pseudocode Representation

## Module 1: Insert Contact

// Procedure to insert a new contact into the phonebook

PROCEDURE insertContact

INPUT: name, phoneNumber

DECLARE contact EXISTS = FALSE

// Check if contact already exists in phonebook

FOR EACH contact IN phonebook

IF contact.name = name THEN

EXISTS = TRUE

BREAK

END IF

END FOR

// Add contact to phonebook if it doesn't exist

IF EXISTS = FALSE THEN

CREATE NEW contact

contact.name = name

contact.phoneNumber = phoneNumber

ADD contact TO phonebook

DISPLAY "Contact added successfully." // Display error message if contact already exists

ELSE

DISPLAY "Contact already exists"

END IF

END PROCEDURE

## Module 2: Search Contact

```
// Procedure to search for a contact in the phonebook

PROCEDURE searchContact

INPUT: name

DECLARE contact FOUND = FALSE  // Flag to check if contact is found


// Flag to check if contact is found

FOR EACH contact IN phonebook

IF contact.name = name THEN

FOUND = TRUE // Set flag to true if contact found

DISPLAY contact.name, contact.phoneNumber

BREAK

END IF

END FOR


// Display error message if contact not found

IF FOUND = FALSE THEN

DISPLAY "Contact not found"

END IF

END PROCEDURE
```

## Module 3: Display All Contacts

```
// Procedure to display all contacts in the phonebook

PROCEDURE displayAllContacts

IF phonebook IS EMPTY THEN

DISPLAY "Phonebook is empty."

ELSE

FOR EACH contact IN phonebook
```

```
DISPLAY contact.name, contact.phoneNumber  
END FOR  
END IF  
END PROCEDURE
```

## Module 4: Delete Contact

```
// Procedure to delete a contact from the phonebook  
PROCEDURE deleteContact  
INPUT: name  
DECLARE contact FOUND = FALSE  
  
// Search for contact in phonebook and delete if found  
FOR EACH contact IN phonebook  
IF contact.name = name THEN  
FOUND = TRUE  
REMOVE contact FROM phonebook  
DISPLAY "Contact deleted successfully."  
BREAK  
END IF  
END FOR  
  
// Display error message if contact not found  
IF FOUND = FALSE THEN  
DISPLAY "Contact not found"  
END IF  
END PROCEDURE
```

## Module 5: Update Contact

```
// Procedure to update a contact's phone number

PROCEDURE updateContact
INPUT: name, newPhoneNumber
DECLARE contact FOUND = FALSE

// Search for contact in phonebook and update if found
FOR EACH contact IN phonebook
IF contact.name = name THEN
FOUND = TRUE
contact.phoneNumber = newPhoneNumber
DISPLAY "Contact updated successfully."
BREAK
END IF
END FOR

// Display error message if contact not found
IF FOUND = FALSE THEN
DISPLAY "Contact not found"
END IF
END PROCEDURE
```

## Module 6 : Sort Contacts

```
// Procedure to sort contacts in ascending order by name

PROCEDURE sortContacts
SORT phonebook BY name IN ASCENDING ORDER
DISPLAY "Contacts sorted successfully."
END PROCEDURE
```

## Module 7: Main Program

```
// Main program loop

PROCEDURE mainProgram

INIT phonebook = EMPTY // Initialize empty phonebook


WHILE TRUE

DISPLAY "Phonebook Application"

DISPLAY "1. Insert Contact"

DISPLAY "2. Search Contact"

DISPLAY "3. Display All Contacts"

DISPLAY "4. Delete Contact"

DISPLAY "5. Update Contact"

DISPLAY "6. Sort Contacts"

DISPLAY "7. Exit"


DISPLAY "Enter your choice: "

GET choice FROM user


// Call corresponding procedure based on user choice

SWITCH choice

CASE 1: CALL insertContact

CASE 2: CALL searchContact

CASE 3: CALL displayAllContacts

CASE 4: CALL deleteContact

CASE 5: CALL updateContact

CASE 6: CALL sortContacts

CASE 7: EXIT PROGRAM

END WHILE

END PROCEDURE
```