Exercise 5

Hariesh R - 23110344

Aim:

To demonstrate string manipulation using an ArrayList, including appending, inserting, searching, filtering by starting letter, and sorting strings in ascending and descending order.

Algorithm:

1. Initialize:
    a. Create an ArrayList<String> to store strings.
    b. Use a Scanner object to handle user input.

2. Display Menu:
    a. Continuously display a menu with options to perform various string operations until the user chooses to exit.

3. Handle User Choice:

    a. Append a string:
        i. Prompt the user to enter a string and add it to the end of the ArrayList.

    b. Insert a string at a specific index:
        i. Prompt the user for an index and a string, then insert the string at the given index in the ArrayList.

    c. Search for a string:
        i. Prompt the user for a string and search the ArrayList for it, returning the index if found, or -1 if not.

    d. List strings starting with a given letter:
        i. Prompt the user for a letter and print all strings in the ArrayList that start with that letter.

    e. Sort strings:
        i. Prompt the user to choose between ascending or descending order, sort the ArrayList accordingly, and print the sorted list.

f. Print the list:
    i. Print all strings currently in the ArrayList.

g. Exit:
    i. Exit the program.

4. Loop until Exit:
    a. Continue to display the menu and handle user input until the user chooses the "Exit" option.

5. Terminate Program:
    a. Close the Scanner and end the program when the user selects "Exit."

Source Code:

```
package Exercise5;
import java.util.ArrayList;
import java.util.Scanner;
import java.util.Collections;

class methods{


   public static void printMenu(){

      System.out.println("\n1. Append");
      System.out.println("2. Insert");
      System.out.println("3. Search");
      System.out.println("4. List all string starts with given letter");
      System.out.println("5. Sort the strings in ascending and descending order");
      System.out.println("6. Print the list");
      System.out.println("7. Exit");
```

```
        System.out.print("Enter: ");
    }

    public static void addElement(ArrayList<String> stringArrayList, String dataString){

        stringArrayList.add(dataString);
    }


    public static void insertElement(ArrayList<String> stringArrayList, String dataString, int
index){

        stringArrayList.add(index, dataString);
    }

    public static int search(ArrayList<String> stringArrayList, String dataString){

        int index = 0;

        for (String stringElement : stringArrayList) {

            if(stringElement.equals(dataString)) return index;
            index++;
        }

        return -1;
    }

    public static void startsWithLetter(ArrayList<String> stringArrayList, String dataString){

        for (String stringElement : stringArrayList) {
```

```
            if(stringElement.startsWith(dataString)){

                System.out.println(stringElement);
            }
        }
    }

    public static ArrayList<String> sortStrings(ArrayList<String> stringArrayList, boolean ascending){

        ArrayList<String> newArrayList = new ArrayList<String>();
//        ArrayList<String> newArrayList = new ArrayList<String>(stringArrayList);

        for (String element: stringArrayList) newArrayList.add(element);

        if(ascending)
            Collections.sort(newArrayList);

        else
            Collections.sort(newArrayList, Collections.reverseOrder());

        return newArrayList;
    }

    public static void printList(ArrayList<String> stringArrayList){

        for (String iString : stringArrayList) {
            System.out.print(iString + " ");
        }
    }
```

```
}

public class main_program {

    static int choice;
    static String data;
    static int index;

    public static void main(String[] args){

        Scanner scanner = new Scanner(System.in);

        ArrayList<String> list = new ArrayList<String>();

        while (true) {

            methods.printMenu();
            choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {

                case 1:
                    System.out.print("Enter the value: ");
                    data = scanner.nextLine();
                    methods.addElement(list, data);

                    break;

                case 2:
```

```
        System.out.print("Enter the index: ");
        index = scanner.nextInt();
        scanner.nextLine();
        System.out.print("Enter the value: ");
        data = scanner.nextLine();
        methods.insertElement(list, data, index);

        break;

    case 3:
        System.out.print("Enter the value: ");
        data = scanner.nextLine();
        int result = methods.search(list, data);
        System.out.print("Found at: " + result);
        break;

    case 4:
        System.out.print("Enter the value: ");
        data = scanner.nextLine();
        methods.startsWithLetter(list, data);
        break;

    case 5:
        System.out.println("1. Ascending");
        System.out.println("2. Descending");
        System.out.print("Enter: ");
        int choice2 = scanner.nextInt();

        if (choice2 == 1){
```

```
            ArrayList<String> newArray = new ArrayList<>();
            newArray = methods.sortStrings(list, true);
            System.out.print("Sorted Array: ");
            methods.printList(newArray);
        }

        else if(choice2 == 2){

            ArrayList<String> newArray = new ArrayList<>();
            newArray = methods.sortStrings(list, false);
            System.out.print("Sorted Array: ");
            methods.printList(newArray);
        }

        else
            System.out.println("Invalid Choice");

        break;

    case 6:
        for (String iString : list) {
            System.out.print(iString + " ");
        }
        break;

    case 7:
        System.out.println("Exited !");
        scanner.close();
        return;
```

```
            default:
                System.out.println("Invalid Choice");
                break;
        }
    }
  }
}
```

Output:

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 1
Enter the value: hariesh
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 1
Enter the value: apple
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 1
Enter the value: zebra
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 1
Enter the value: banana
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 6
hariesh apple zebra banana
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 2
Enter the index: 2
Enter the value: hello
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 6
hariesh apple hello zebra banana
```

```
hariesh apple hello zebra banana
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 3
Enter the value: apple
Found at: 1
```

```
Found at: 1
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 3
Enter the value: hey
Found at: -1
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 4
Enter the value: a
apple
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 5
1. Ascending
2. Descending
Enter: 1
Sorted Array: apple banana hariesh hello zebra
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 5
1. Ascending
2. Descending
Enter: 2
Sorted Array: zebra hello hariesh banana apple
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 6
hariesh apple hello zebra banana
```

```
1. Append
2. Insert
3. Search
4. List all string starts with given letter
5. Sort the strings in ascending and descending order
6. Print the list
7. Exit
Enter: 7
Exited !

Process finished with exit code 0
```

Result:

This program lets users manipulate an ArrayList of strings through a menu, allowing operations like appending, inserting, searching, filtering by starting letter, sorting, and printing. For example, users can create and modify a list like ["apple", "banana", "cherry"] and view the updated list after each operation before exiting the program.