FILE STRUCTURES LABORATORY WITH MINI PROJECT

Subject Code: 18ISL68 **I.A. Marks : 20**
Hours/Week : 01I+02P **Exam Hours: 03**
Total Hours :40 **Exam Marks: 80**

PART – A

Design, develop, and implement the following programs

1. Write a program to read series of names, one per line, from standard input and write these names spelled in reverse order to the standard output using I/O redirection and pipes. Repeat the exercise using an input file specified by the user instead of the standard input and using an output file specified by the user instead of the standard output.

2. Write a program to read and write student objects with fixed-length records and the fields delimited by "|". Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

3. Write a program to read and write student objects with Variable - Length records using any suitable record structure. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

4. Write a program to write student objects with Variable - Length records using any suitable record structure and to read from this file a student record using RRN.

5. Write a program to implement simple index on primary key for a file of student objects. Implement add ( ), search ( ), delete ( ) using the index.

6. Write a program to implement index on secondary key, the name, for a file of student objects. Implement add( ), search ( ), delete ( ) using the secondary index.

7. Write a program to read two lists of names and then match the names in the two lists using Consequential Match based on a single loop. Output the names common to both the lists.

8. Write a program to read 'k' lists of names and merge them using k-way merge algorithm with k=8

1. Write a program to read series of names, one per line, from standard input and write these names spelled in reverse order to the standard output using I/O redirection and pipes. Repeat the exercise using an input file specified by the user instead of the standard input and using an output file specified by the user instead of the standard output.

```
import java.util.*;
import java.io.*;

class Lab1
{
	Scanner scan = new Scanner(System.in);
	public static void main(String args[]) throws IOException
	{
		Lab1 obj = new Lab1();
		int choice;
		while(true)
		{
			System.out.println("**********************");
			System.out.println("1. Accept Input from Standard Input Device");
			System.out.println("2. Accept Input from File");
			System.out.println("3. Exit");
			System.out.println("**********************");

			System.out.println("Please enter your choice:");
			choice = obj.scan.nextInt();
			switch(choice)
			{
			case 1:
				obj.StdInput();
				break;
			case 2:
				obj.FileInput();
				break;
			case 3:
				System.out.println("You chose exit!");
				System.exit(0);
			default:
				System.out.println("Invalid Option");
			}
		}
	}
	void StdInput()
	{
		String org, rev = "";
		int n;
		System.out.println("Enter the number of names to be Reversed");
		n = scan.nextInt();
```

```
            scan.nextLine();
            while(n!=0)
            {
                  System.out.println("Enter the  name to be Reversed");
                  org = scan.nextLine();
                  rev= Rev_String(org);
                  System.out.println("Reverse of entered string is: "+rev);
                  n=n-1;
            }
      }

      void FileInput()throws IOException
      {
            System.out.println("Enter the input file name(with extension)");
            String infile = scan.next();
            System.out.println("Enter the output file name(with extension)");
            String outfile = scan.next();
            String org, rev = "";

            BufferedReader br = new BufferedReader(new FileReader(infile));
            PrintWriter pw = new PrintWriter(outfile);

            while((org = br.readLine()) != null)
            {
                  rev= Rev_String(org);
                  pw.println(rev);
            }
            pw.flush();
            System.out.println("All Reverse Names written to " + outfile);

            br.close();
            pw.close();
      }

      String Rev_String(String org)
      {
            StringBuilder sb = new StringBuilder(org);
            sb.reverse();
            return sb.toString();
      }
}
```

Output:

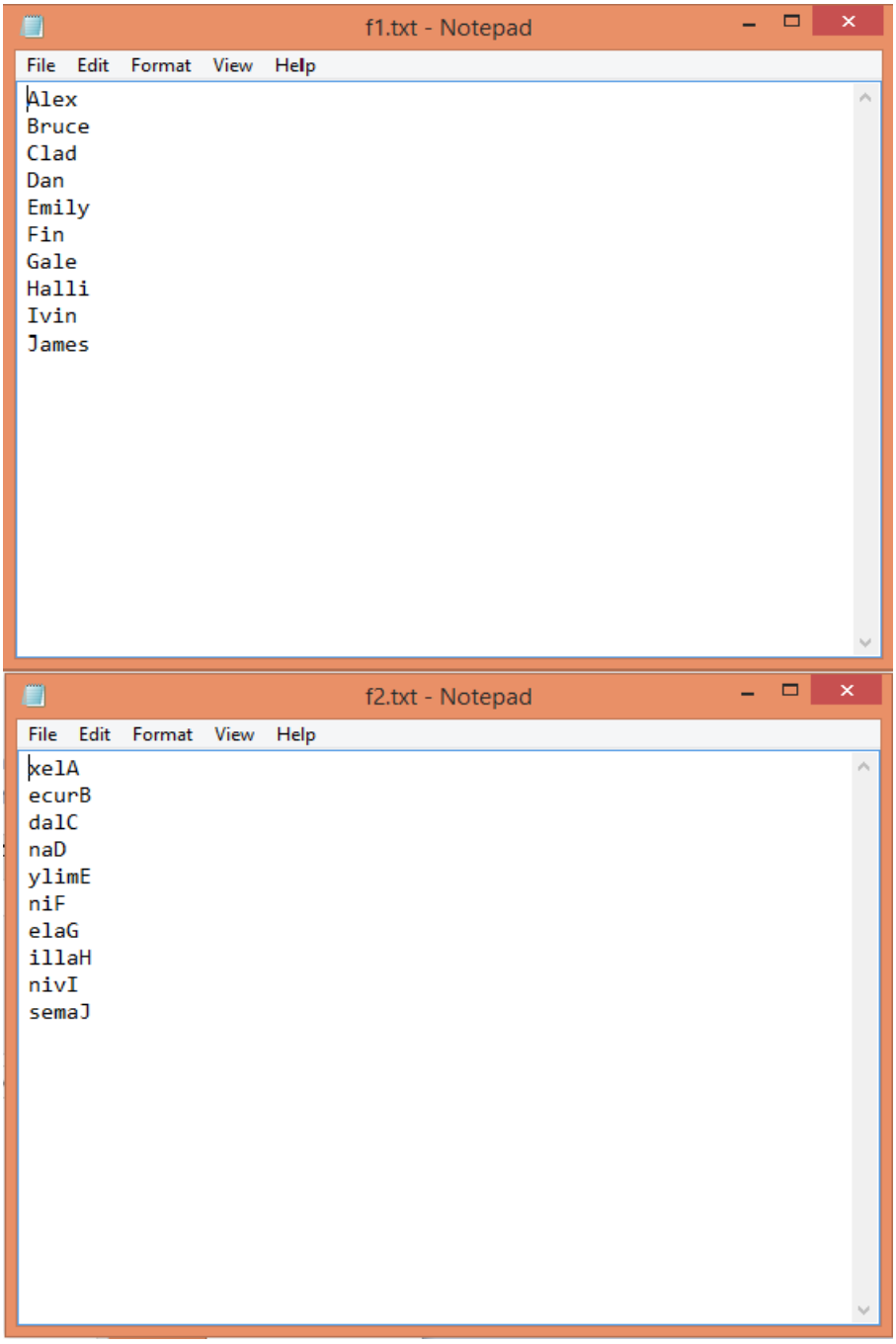
****

1. Accept Input from Standard Input Device

2. Accept Input from File
3. Exit
****
Please enter your choice:
1
Enter the number of names to be Reversed
2
Enter the name to be Reversed
Tom
Reverse of entered string is: moT
Enter the name to be Reversed
Jerry
Reverse of entered string is: yrreJ

****
1. Accept Input from Standard Input Device
2. Accept Input from File
3. Exit
****
Please enter your choice:
2
Enter the input file name(with extension)
f1.txt
Enter the output file name(with extension)
f2.txt
All Reverse Names written to f2.txt

****
1. Accept Input from Standard Input Device
2. Accept Input from File
3. Exit
****
Please enter your choice:
3
You chose exit!


f1.txt - Notepad
File Edit Format View Help
Alex
Bruce
Clad
Dan
Emily
Fin
Gale
Halli
Ivin
James
f2.txt - Notepad
File Edit Format View Help
kelA
ecurB
dalC
naD
ylimE
niF
elaG
illaH
nivI
semaJ

2. Write a program to read and write student objects with fixed-length records and the fields delimited by "|". Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

```
import java.io.*;
import java.util.*;

class Lab2
{
	final int size=50;
	Scanner scan = new Scanner(System.in);
	public static void main(String[] args) throws IOException,NullPointerException
	{
		Lab2 obj = new Lab2();
		int choice;
		while(true)
		{
			System.out.println("**********************");
			System.out.println("1.Pack()");
			System.out.println("2.Unpack()");
			System.out.println("3.Search()");
			System.out.println("4.Modify()");
			System.out.println("5.Exit");
			System.out.println("**********************");
			System.out.println("Please enter your choice:");
			choice = obj.scan.nextInt();
			obj.scan.nextLine();
			switch(choice)
			{
			case 1:
				obj.pack();
				break;
			case 2:
				obj.unpack();
				break;
			case 3:
				obj.search();
				break;
			case 4:
				obj.modify();
				break;
			case 5:
				System.out.println("You chose exit!");
				System.exit(0);
			default:
				System.out.println("Invalid Option");
			}
```

```
                }
        }

        public void pack() throws FileNotFoundException
        {
                System.out.println("Enter Name, USN, Sem and Branch ");
                String name = scan.nextLine();
                String usn = scan.nextLine();
                String sem = scan.nextLine();
                String branch = scan.nextLine();
                PrintWriter pw = new PrintWriter(new FileOutputStream(new
File("student.txt"),true));
                String b = name + "|" + usn + "|" + sem + "|" + branch + "|";
                int len = b.length();
                String s1 = "-";
                if(len<50)
                {
                        for(int j=len;j<=50;j++)
                        b = b.concat(s1);
                }
                pw.println(b);
                pw.flush();
                pw.close();
        }

        public void unpack()throws IOException
        {
                String name = "" ,usn = "" ,sem = "" ,branch = "", s;
                BufferedReader br = new BufferedReader(new FileReader("student.txt"));
                while((s = br.readLine())!=null)
                {
                        String[] result = s.split("\\|");
                        name = result[0];
                        usn = result[1];
                        sem = result[2];
                        branch = result[3];
                        System.out.println("The details are: " + name + " " + usn + " " + sem + " " +
branch);
                }
                br.close();
        }

        public void search()throws FileNotFoundException, IOException
        {
                BufferedReader br = new BufferedReader(new FileReader("student.txt"));
                String name = "", usn = "", sem = "", branch = "", r;
```

```
            System.out.println("Enter the usn");
            String usn1 = scan.nextLine();
            while((r= br.readLine()) !=null)
            {
                  String[] result = r.split("\\|");
                  name=result[0];
                  usn=result[1];
                  sem= result[2];
                  branch=result[3];
                  if(usn.equals(usn1))
                  {
                        System.out.println("Match found. The details of the record are:");
                        System.out.println(name + " " + usn + " " + sem + " " + branch);
                        br.close();
                        return;
                  }
            }
            System.out.println("Record not found");
            br.close();
      }

      public void modify() throws FileNotFoundException,IOException,NullPointerException
      {
            String name = "", usn = "", sem = "", branch = "", r;
            File file = new File("student.txt");
            BufferedReader br = new BufferedReader(new FileReader(file));
            File temp = new File("temp.txt");
            PrintWriter pw = new PrintWriter(temp);
            System.out.println("Enter usn");
            String usn1 = scan.nextLine();

            while((r= br.readLine()) !=null)
            {
                  String[] result = r.split("\\|");
                  name=result[0];
                  usn=result[1];
                  sem= result[2];
                  branch=result[3];
                  if(usn.equals(usn1))
                  {
                        System.out.println("The details are: " + name + " " + usn + " " + sem +
" " + branch);
                        System.out.println("enter name, usn,sem and branch");
                        String name11 = scan.nextLine();
                        String usn11 = scan.nextLine();
                        String sem11 = scan.nextLine();
```

```
                                String branch11 = scan.nextLine();
                                String b = name11+"|"+usn11+"|"+sem11+"|"+branch11+"|";
                                int le = b.length();
                                String s1 = "-";
                                if(le<50)
                                {
                                        for(int j=le;j<=50;j++)
                                                b = b.concat(s1);
                                        pw.println(b);
                                }
                        }
                        else
                        {
                                pw.println(r);
                        }
                }
                pw.flush();
                pw.close();
                br.close();
                file.delete()
                temp.renameTo(file)
                System.out.println("File Modified");
        }
}
```

Output :

```
**********************
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********************
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Moahn
1HK15CV030
06
Civil
**********************
1.Pack()
2.Unpack()
3.Search()
4.Modify()
```

```
5.Exit
**********
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Allan Jones
1HK15ME013
06
Mechanical
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Suraj Kumar
1HK15EE052
06
Electrical and Electronics
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Deepak Joshua
1HK15EC010
06
Electronics and Communication
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
2
```

The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electroni
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Commun
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
3
Enter the usn
1HK15IS001
Record not found
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
3
Enter the usn
1HK15EE052
Match found. The details of the record are:
Suraj Kumar 1HK15EE052 06 Electrical and Electroni
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
4
Enter usn
1HK15EC010
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Commun
enter name, usn,sem and branch
Deepak Joshua
1HK15EC010
06
ECE

File Modified
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
2
The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electroni
The details are: Deepak Joshua 1HK15EC010 06 ECE
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
5
You chose exit!

3. **Write a program to read and write student objects with Variable - Length records using any suitable record structure. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.**

```
import java.io.*;
import java.util.*;

class Lab3
{
	Scanner scan = new Scanner(System.in);

	public static void main(String[] args) throws IOException,NullPointerException
	{
		Lab2 obj = new Lab2();
		int choice;
		while(true)
		{
			System.out.println("**********************");
			System.out.println("1.Pack()");
			System.out.println("2.Unpack()");
			System.out.println("3.Search()");
			System.out.println("4.Modify()");
			System.out.println("5.Exit");
			System.out.println("**********************");
			System.out.println("Please enter your choice:");
			choice = obj.scan.nextInt();
			obj.scan.nextLine();
			switch(choice)
			{
			case 1:
				obj.pack();
				break;
			case 2:
				obj.unpack();
				break;
			case 3:
				obj.search();
				break;
			case 4:
				obj.modify();
				break;
			case 5:
				System.out.println("You chose exit!");
				System.exit(0);
			default:
				System.out.println("Invalid Option");
			}
		}
```

```
	}

	public void pack() throws FileNotFoundException
	{
		System.out.println("Enter Name, USN, Sem and Branch ");
		String name = scan.nextLine();
		String usn = scan.nextLine();
		String sem = scan.nextLine();
		String branch = scan.nextLine();
		PrintWriter pw = new PrintWriter(new FileOutputStream(new
File("student.txt"),true));
		String b = name + "|" + usn + "|" + sem + "|" + branch + "|";

		pw.println(b);

		pw.flush();
		pw.close();
	}

	public void unpack()throws IOException
	{
		String name = "" ,usn = "" ,sem = "" ,branch = "", s;
		BufferedReader br = new BufferedReader(new FileReader("student.txt"));
		while((s = br.readLine())!=null)
		{
			String[] result = s.split("\\|");
			name = result[0];
			usn = result[1];
			sem = result[2];
			branch = result[3];
			System.out.println("The details are: " + name + " " + usn + " " + sem + " " +
branch);
		}
		br.close();
	}

	public void search()throws FileNotFoundException, IOException
	{
		BufferedReader br = new BufferedReader(new FileReader("student.txt"));
		String name = "", usn = "", sem = "", branch = "", r;
		System.out.println("Enter the usn");
		String usn1 = scan.nextLine();
		while((r= br.readLine()) !=null)
		{
			String[] result = r.split("\\|");
			name=result[0];
```

```
                usn=result[1];
                sem= result[2];
                branch=result[3];
                if(usn.equals(usn1))
                {
                        System.out.println("Match found. The details of the record are:");
                        System.out.println(name + " " + usn + " " + sem + " " + branch);
                        br.close();
                        return;
                }
        }
        System.out.println("Record not found");
        br.close();
}

public void modify() throws FileNotFoundException,IOException,NullPointerException
{
        String name = "", usn = "", sem = "", branch = "", r;
        File file = new File("student.txt");
        BufferedReader br = new BufferedReader(new FileReader(file));
        File temp = new File("temp.txt");
        PrintWriter pw = new PrintWriter(temp);
        System.out.println("Enter usn");
        String usn1 = scan.nextLine();

        while((r= br.readLine()) !=null)
        {
                String[] result = r.split("\\|");
                name=result[0];
                usn=result[1];
                sem= result[2];
                branch=result[3];

                if(usn.equals(usn1))
                {
                        System.out.println("The details are: " + name + " " + usn + " " + sem +
" " + branch);
                        System.out.println("enter name, usn,sem and branch");
                        String name11 = scan.nextLine();
                        String usn11 = scan.nextLine();
                        String sem11 = scan.nextLine();
                        String branch11 = scan.nextLine();
                        String b = name11+"|"+usn11+"|"+sem11+"|"+branch11+"|";
                        int le = b.length();

                        String s1 = "-";
```

```
                        if(le<50)
                        {
                                for(int j=le;j<=50;j++)
                                        b = b.concat(s1);
                                pw.println(b);
                        }
                }
                else
                {
                        pw.println(r);
                }
        }
        pw.flush();
        pw.close();
        br.close();

        file.delete()
        temp.renameTo(file)
        System.out.println("File Modified");
    }
}
```

Output:

```
**********************
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********************
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Moahn
1HK15CV030
06
Civil
**********************
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********************
Please enter your choice:
1
```

```
Enter Name, USN, Sem and Branch
Allan Jones
1HK15ME013
06
Mechanical
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Suraj Kumar
1HK15EE052
06
Electrical and Electronics
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
1
Enter Name, USN, Sem and Branch
Deepak Joshua
1HK15EC010
06
Electronics and Communication
**********
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
**********
Please enter your choice:
2
The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electronics
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Communication
```

The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electronics
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Communication
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
3
Enter the usn
1HK15EC010
Match found. The details of the record are:
Deepak Joshua 1HK15EC010 06 Electronics and Communication
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
3
Enter the usn
1HK15CS400
Record not found
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
4
Enter usn
1HK15EC010
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Communication
enter name, usn,sem and branch
Deepak Joshua
1HK16EC010
04
ECE

File Modified
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
2
The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electronics
The details are: Deepak Joshua 1HK16EC010 04 ECE
****
1.Pack()
2.Unpack()
3.Search()
4.Modify()
5.Exit
****
Please enter your choice:
5
You chose exit!

4. Write a program to write student objects with Variable - Length records using any suitable record structure and to read from this file a student record using RRN.

```
import java.io.*;
import java.util.*;

class Lab4
{
	public static int count;
	public static final int[] rrn = new int[20];
	Scanner scan = new Scanner(System.in);

	public static void main(String args[])throws IOException
	{
		Lab4 obj = new Lab4();
		obj.createrrn();

		int choice;

		while(true)
		{
			System.out.println("**********************");
			System.out.println("1. Pack()");
			System.out.println("2. Unpack()");
			System.out.println("3. Search()");
			System.out.println("4. Exit");
			System.out.println("**********************");
			System.out.println("Please enter your choice:");
			choice = obj.scan.nextInt();
			obj.scan.nextLine();
			switch(choice)
			{
			case 1:
				obj.pack();
				break;
			case 2:
				obj.unpack();
				break;
			case 3:
				System.out.println("Enter the rrn number to search the record");
				int r = obj.scan.nextInt();
				obj.search(r);
				break;
			case 4:
				System.out.println("You chose exit!");
				System.exit(0);
			default:
```

```
                        System.out.println("Invalid option");
                        break;
                }
        }
}

public void createrrn()throws IOException
{
        count = -1;
        long pos;
        RandomAccessFile file = new RandomAccessFile("student1.txt", "r");
        pos = file.getFilePointer();
        String s;
        while((s = file.readLine())!=null)
        {
                count++;
                rrn[count] = (int)pos;
                pos = file.getFilePointer();
                String[] result = s.split("\\|");
                String name = result[0];
                System.out.println("The rrn for " + name + " is " + count);
        }
        file.close();
}

public void pack() throws IOException
{
        System.out.println("Enter Name,USN,Sem and Branch ");
        String name = scan.nextLine();
        String usn = scan.nextLine();
        String sem = scan.nextLine();
        String branch = scan.nextLine();
        PrintWriter pw = new PrintWriter(new FileOutputStream(new
File("student1.txt"),true));
        String b = name+"|"+usn+"|"+sem+"|"+branch+"|"+"$";
        pw.println(b);
        pw.flush();
        pw.close();
        createrrn();
}

public void unpack()throws IOException
{
        String name = "" ,usn = "" ,sem = "" ,branch = "", s;
        BufferedReader br = new BufferedReader(new FileReader("student1.txt"));
        while((s = br.readLine())!=null)
```

```
                {
                        String[] result = s.split("\\|");
                        name = result[0];
                        usn = result[1];
                        sem = result[2];
                        branch = result[3];
                        System.out.println("The details are: " + name + " " + usn + " " + sem + " " +
branch);
                }
                br.close();
        }

        public void search(int x)throws IOException
        {
                RandomAccessFile file = new RandomAccessFile("student1.txt", "rw");
                if(x>count)
                {
                        System.out.println("Record not found");
                        file.close();
                        return;
                }
                else
                {
                        int pos = rrn[x];
                        file.seek(pos);
                        String a = file.readLine();
                        System.out.println("the Record is :"+a);
                }
                file.close();
        }
}
```

Output:

```
**********************
1. Pack()
2. Unpack()
3. Search()
4. Exit
**********************
Please enter your choice:
1
Enter Name,USN,Sem and Branch
Moahn
1HK15CV030
06
Civil
```

The rrn for Moahn is 0
****
1. Pack()
2. Unpack()
3. Search()
4. Exit
****
Please enter your choice:
1
Enter Name,USN,Sem and Branch
Allan Jones
1HK15ME013
06
Mechanical
The rrn for Moahn is 0 The
rrn for Allan Jones is 1
****
1. Pack()
2. Unpack()
3. Search()
4. Exit
****
Please enter your choice:
1
Enter Name,USN,Sem and Branch
Suraj Kumar
1HK15EE052
06
Electrical and Electronics
The rrn for Moahn is 0 The
rrn for Allan Jones is 1 The
rrn for Suraj Kumar is 2
****
1. Pack()
2. Unpack()
3. Search()
4. Exit
****
Please enter your choice:
1
Enter Name,USN,Sem and Branch
Deepak Joshua
1HK15EC010
06
Electronics and Communication
The rrn for Moahn is 0

```
The rrn for Allan Jones is 1
The rrn for Suraj Kumar is 2
The rrn for Deepak Joshua is 3
**********
1. Pack()
2. Unpack()
3. Search()
4. Exit
**********
Please enter your choice:
2
The details are: Moahn 1HK15CV030 06 Civil
The details are: Allan Jones 1HK15ME013 06 Mechanical
The details are: Suraj Kumar 1HK15EE052 06 Electrical and Electronics
The details are: Deepak Joshua 1HK15EC010 06 Electronics and Communication
**********
1. Pack()
2. Unpack()
3. Search()
4. Exit
**********
Please enter your choice:
3
Enter the rrn number to search the record
2
the Record is :Suraj Kumar|1HK15EE052|06|Electrical and Electronics|$
**********
1. Pack()
2. Unpack()
3. Search()
4. Exit
**********
Please enter your choice:
3
Enter the rrn number to search the record
5
Record not found
**********
1. Pack()
2. Unpack()
3. Search()
4. Exit
**********
Please enter your choice:
4
You chose exit!
```

5. Write a program to implement simple index on primary key for a file of student objects. Implement add ( ), search ( ), delete ( ) using the index.

```
import java.io.*;
import java.util.Scanner;

public class Lab5
{
	public static int count;
	public static final int[] Address_list = new int[100];
	public static final String[] usn_list = new String[100];
	public static Scanner s = new Scanner(System.in);

	public static void main(String[] args)throws IOException
	{
		Lab5 obj = new Lab5();
		obj.create_index();
		int ch;
		System.out.println("******Menu******");
		System.out.println("1. Add Record");
		System.out.println("2. Search Record");
		System.out.println("3. Remove Record");
		System.out.println("4. Exit");
		System.out.println("****************");
		while(true)
		{
			System.out.println("\nPlease enter your choice:");
			ch = s.nextInt();
			s.nextLine();
			switch(ch)
			{
			case 1:
				obj.insert();
				break;
			case 2:
				obj.search();
				break;
			case 3:
				obj.remove();
				break;
			case 4:
				System.out.println("Do you want to exit? (Y/N)");
				if(s.next().equalsIgnoreCase("y"))
				{
					System.out.println("Program Ended");
					System.exit(0);
				}
```

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

public void create_index()throws IOException,ArrayIndexOutOfBoundsException
{
        count = -1;
        long pos;
        RandomAccessFile file = new RandomAccessFile("f1.txt", "r");
        pos = file.getFilePointer();
        String s ;
        while((s = file.readLine())!=null)
        {
                String[] result = s.split("\\|");
                count++;
                usn_list[count] = result[0];
                Address_list[count] = (int)pos;
                pos=file.getFilePointer();
        }
        file.close();
        sort_index();
}

public void sort_index()throws IOException
{
        for(int i=0;i<=count;i++)
        {
                for(int j=i+1;j<=count;j++)
                {
                        if(usn_list[i].compareTo(usn_list[j])>0)
                        {
                                String temp = usn_list[i];
                                usn_list[i] = usn_list[j];
                                usn_list[j] = temp;
                                int temp1 = Address_list[i];
                                Address_list[i]=Address_list[j];
                                Address_list[j]=temp1;
                        }
                }
        }
}

public void insert()throws IOException,FileNotFoundException
```

```
{
        PrintWriter pw = new PrintWriter(new FileOutputStream(new File("f1.txt"),true));
        System.out.println("Enter USN,Name,Sem and Branch ");
        String usn = s.nextLine();
        String name = s.nextLine();
        String sem = s.nextLine();
        String branch = s.nextLine();
        String b = usn+"|"+name+"|"+sem+"|"+branch+"|"+"$";
        pw.println(b);
        pw.close();
        create_index();
}

public void search()throws IOException
{
        int pos;
        System.out.println("Enter the usn to be searched");
        String key = s.nextLine();
        pos = search_index(key);

        if(pos!=-1)
                display_record(pos);
        else
                System.out.println("Record not found");
}

public int search_index(String key)
{
        int low = 0, high = count, mid = 0;
        while(low <= high)
        {
                mid = (low + high)/2;
                if(usn_list[mid].equals(key))
                        return mid;

                if(usn_list[mid].compareTo(key)>0)
                        high = mid - 1;

                if(usn_list[mid].compareTo(key)<0)
                        low = mid + 1;
        }
        return -1;
}

public void display_record(int pos)throws IOException
{
```

```
        RandomAccessFile file = new RandomAccessFile("f1.txt", "r");

        int address = Address_list[pos];
        String usn="",sem="",branch="",name="";

        file.seek(address);
        String s = file.readLine();

        while(s!=null)
        {
                String[] result = s.split("\\|");
                usn = result[0];
                name = result[1];
                sem = result[2];
                branch = result[3];
                System.out.println("\nRecord Details");
                System.out.println("USN: " + usn);
                System.out.println("Name: " + name);
                System.out.println("Sem: " + sem);
                System.out.println("Branch: " + branch);
                break;
        }
        file.close();
}

public void remove()throws IOException
{
        System.out.println("Enter the key to be deleted");
        String key = s.nextLine();

        int pos = search_index(key);
        if(pos != -1)
        {
                delete_from_file(pos);
                create_index();
        }
        else
                System.out.println("Record not found");
}

public void delete_from_file(int pos)throws IOException
{
        display_record(pos);

        RandomAccessFile file = new RandomAccessFile("f1.txt", "rw");
        System.out.println("Are you sure you want to delete? (Y/N)");
```

```
                String ch = s.nextLine();

                if(ch.equalsIgnoreCase("y"))
                {
                        int address= Address_list[pos];
                        String del_ch="*";
                        file.seek(address);
                        file.writeBYtes(del_ch);
                        System.out.println("Record is deleted");
                }
                file.close();
        }
}
```

Output:

******Menu******
1. Add Record
2. Search Record
3. Remove Record
4. Exit
****

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15CV030
Moahn
06
Civil

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15ME013
Allan Jones
06
Mechanical

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15EE052
Suraj Kumar
06
Electrical and Electronics

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15EC010
Deepak Joshua
06
Electronics and Communication

Please enter your choice:
2
Enter the usn to be searched
1HK15ME013

Record Details
USN: 1HK15ME013
Name: Allan Jones
Sem: 06
Branch: Mechanical

Please enter your choice:
3
Enter the key to be deleted
1HK15ME013

Record Details
USN: 1HK15ME013
Name: Allan Jones
Sem: 06
Branch: Mechanical
Are you sure you want to delete? (Y/N)
y
Record is deleted

Please enter your choice:
2
Enter the usn to be searched
1HK15ME013
Record not found

Please enter your choice:
4
Do you want to exit? (Y/N)
Y
Program Ended

6. Write a program to implement index on secondary key, the name, for a file of student objects. Implement add ( ), search ( ), delete ( ) using the secondary index.

```
import java.io.*;
import java.util.Scanner;

public class Lab6
{
	public static int count;
	public static final int[] Address_list = new int[100];
	public static final String[] Name_list = new String[100];
	public static Scanner s = new Scanner(System.in);

	public static void main(String[] args)throws IOException
	{
		Lab6 obj = new Lab6();
		obj.create_index();
		int ch;
		System.out.println("******Menu******");
		System.out.println("1. Add Record");
		System.out.println("2. Search Record");
		System.out.println("3. Remove Record");
		System.out.println("4. Exit");
		System.out.println("****************");
		while(true)
		{
			System.out.println("\nPlease enter your choice:");
			ch = s.nextInt();
			s.nextLine();
			switch(ch)
			{
			case 1:
				obj.insert();
				break;
			case 2:
				obj.search();
				break;
			case 3:
				obj.remove();
				break;
			case 4:
				System.out.println("Do you want to exit? (Y/N)");
				if(s.next().equalsIgnoreCase("y"))
				{
					System.out.println("Program Ended");
					System.exit(0);
				}
```

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

public void create_index()throws IOException,ArrayIndexOutOfBoundsException
{
        count = -1;
        long pos;
        RandomAccessFile file = new RandomAccessFile("f1.txt", "r");
        pos = file.getFilePointer();
        String s ;
        while((s = file.readLine())!=null)
        {
                String[] result = s.split("\\|");
                count++;
                Name_list[count] = result[1];
                Address_list[count] = (int)pos;
                pos=file.getFilePointer();
        }
        file.close();
        sort_index();
}

public void sort_index()throws IOException
{
        for(int i=0;i<=count;i++)
        {
                for(int j=i+1;j<=count;j++)
                {
                        if(Name_list[i].compareTo(Name_list[j])>0)
                        {
                                String temp = Name_list[i];
                                Name_list[i] = Name_list[j];
                                Name_list[j] = temp;

                                int temp1 = Address_list[i];
                                Address_list[i]=Address_list[j];
                                Address_list[j]=temp1;
                        }
                }
        }
}
```

```
public void insert()throws IOException,FileNotFoundException
{
        PrintWriter pw = new PrintWriter(new FileOutputStream(new File("f1.txt"),true));

        System.out.println("Enter USN,Name,Sem and Branch ");
        String usn = s.nextLine();
        String name = s.nextLine();
        String sem = s.nextLine();
        String branch = s.nextLine();
        String b = usn+"|"+name+"|"+sem+"|"+branch+"|"+"$";

        pw.println(b);
        pw.close();

        create_index();
}

public void search()throws IOException
{
        int pos;
        System.out.println("Enter the name to be searched");
        String key = s.nextLine();

        int t = 0;
        pos = search_index(key);

        if(pos!=-1)
        {
                display_record(pos);

                t = pos;
                while((t<count)&&(Name_list[++t].equals(key)))
                        display_record(t);

                t = pos;
                while((t>=0) &&(Name_list[--t].equals(key)))
                        display_record(t);
        }
        else
                System.out.println("Record not found");
}

public int search_index(String key)
{
        int low = 0, high = count, mid = 0;
        while(low <= high)
```

```
        {
                mid = (low + high)/2;
                if(Name_list[mid].equals(key))
                        return mid;

                if(Name_list[mid].compareTo(key)>0)
                        high = mid - 1;

                if(Name_list[mid].compareTo(key)<0)
                        low = mid + 1;
        }
        return -1;
}

public void display_record(int pos)throws IOException
{
        RandomAccessFile file = new RandomAccessFile("f1.txt", "r");

        int address = Address_list[pos];
        String usn="",sem="",branch="",name="";

        file.seek(address);
        String s = file.readLine();

        while(s!=null)
        {
                String[] result = s.split("\\|");
                usn = result[0];
                name = result[1];
                sem = result[2];
                branch = result[3];
                System.out.println("\nRecord Details");
                System.out.println("USN: " + usn);
                System.out.println("Name: " + name);
                System.out.println("Sem: " + sem);
                System.out.println("Branch: " + branch);
                break;
        }
        file.close();
}

public void remove()throws IOException
{
        int pos, t;

        System.out.println("Enter the key to be deleted");
```

```
                String key = s.nextLine();

                pos = search_index(key);
                if(pos != -1)
                {
                        delete_from_file(pos);

                        t = pos;
                        while((t<count)&&(Name_list[++t].equals(key)))
                                delete_from_file(t);

                        t=pos;
                        while((t>=0) &&(Name_list[--t].equals(key)))
                                delete_from_file(t);

                        create_index();
                }
                else
                        System.out.println("Record not found");
        }

        public void delete_from_file(int pos)throws IOException
        {
                display_record(pos);

                RandomAccessFile file = new RandomAccessFile("f1.txt", "rw");
                System.out.println("Are you sure you want to delete? (Y/N)");
                String ch = s.nextLine();

                if(ch.equalsIgnoreCase("y"))
                {
                        int address= Address_list[pos];
                        String del_ch="*";

                        file.seek(address);
                        String str = file.readLine();
                        int x = str.indexOf('|');
                        x++;
                        file.seek(address + x);
                        file.writeBytes(del_ch);
                        System.out.println("Record is deleted");
                }
                file.close();
        }
}
```

Output:

******Menu******
1. Add Record
2. Search Record
3. Remove Record
4. Exit
****

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15CV030
Moahn
06
Civil

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15ME013
Allan Jones
06
Mechanical

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15EE052
Suraj Kumar
06
Electrical and Electronics

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15EC010
Deepak Joshua
06
Electronics and Communication

Please enter your choice:
1
Enter USN,Name,Sem and Branch
1HK15CS026
Moahn
06

Computer Science

Please enter your choice:
2
Enter the name to be searched
Moahn

Record Details
USN: 1HK15CV030
Name: Moahn
Sem: 06
Branch: Civil

Record Details
USN: 1HK15CS026
Name: Moahn
Sem: 06
Branch: Computer Science

Please enter your choice:
3
Enter the key to be deleted
Moahn

Record Details
USN: 1HK15CV030
Name: Moahn
Sem: 06
Branch: Civil
Are you sure you want to delete? (Y/N)
Y
Record is deleted

Record Details
USN: 1HK15CS026
Name: Moahn
Sem: 06
Branch: Computer Science
Are you sure you want to delete? (Y/N)
N

Please enter your choice:
2
Enter the name to be searched
Moahn

Record Details
USN: 1HK15CS026
Name: Moahn
Sem: 06
Branch: Computer Science

Please enter your choice:
4
Do you want to exit? (Y/N)
Y
Program Ended

7. Write a program to read two lists of names and then match the names in the two lists using Cosequential Match based on a single loop. Output the names common to both the lists.

```
import java.util.*;
import java.io.*;

public class Lab7
{
	Scanner scan = new Scanner(System.in);

	public static void main(String[] args)throws IOException
	{
		Lab7 obj = new Lab7();

		System.out.println("Enter the names in list 1 (Enter # to terminate the list)");
		obj.readNames("list1.txt");
		System.out.println("Enter the names in list 2 (Enter # to terminate the list)");

		obj.readNames("list2.txt");

		obj.combineLists();
		obj.display();

	}

	public void readNames(String fname) throws FileNotFoundException
	{
		String s[] = new String[50];
		PrintWriter pw = new PrintWriter(fname);
		int i = 0, j;
		for(i=0;;i++)
		{
			s[i] = scan.nextLine();
			if(s[i].equals("#"))
				break;
		}

		sort(s,i);

		for(j=0;j<i;j++)
			pw.println(s[j]);

		pw.close();
	}

	public void sort(String s[],int count)
	{
```

```
		String temp;
		for(int i=0;i<count;i++)
			for(int j=i+1;j<count;j++)
				if(s[i].compareTo(s[j])>0)
				{
					temp = s[i];
					s[i]=s[j];
					s[j]=temp;
				}
}

public void combineLists()throws IOException
{
		BufferedReader br1 = new BufferedReader(new FileReader("list1.txt"));
		BufferedReader br2 = new BufferedReader(new FileReader("list2.txt"));
		PrintWriter pw = new PrintWriter("list3.txt");

		String name1 = br1.readLine();
		String name2 = br2.readLine();

		while(name1 != null && name2 != null)
		{
			if(name1.equals(name2))
			{
				pw.println(name1);
				name1 = br1.readLine();
				name2 = br2.readLine();
			}

			else if(name1.compareTo(name2)<0)
				name1 = br1.readLine();

			else
				name2 = br2.readLine();
		}

		pw.close();
		br2.close();
		br1.close();
}

public void display()throws IOException
{
		BufferedReader b = new BufferedReader(new FileReader("list3.txt"));
		String l = b.readLine();
		if(l == null)
```

```
                System.out.println("No matching string");
            else
            {
                System.out.println("Common names in both lists are:");
                do
                {
                    System.out.println(l);
                }while((l = b.readLine()) != null);
            }
            b.close();
        }
}
```

Output :

```
Enter the names in list 1 (Enter # to terminate the list)
Suraj
Moahn
Abhijeet
Akask
Vikranth
Praveen
#
Enter the names in list 2 (Enter # to terminate the list)
Abhijeet
Praveen
Prakash
Vikranth
Nisha
Anisha
Sanjay
#
Common names in both lists are:
Abhijeet
Praveen
Vikranth
```

8. Write a program to read k Lists of names and merge them using k-way merge algorithm with k = 8.

```
import java.io.*;
import java.util.*;

public class Lab8
{
	int k = 8;
	public static void main(String[] args)throws FileNotFoundException, IOException
	{
		Lab8 m = new Lab8();
		m.create();
		m.mergeFiles();
		m.display();
	}

	public void create()throws FileNotFoundException
	{
		Scanner scan = new Scanner(System.in);
		for(int i = 1; i <= k; i++)
		{
			System.out.println("Enter the of names in list " + i + ". Enter # to terminate
list");
			PrintWriter pw = new PrintWriter("list"+i+".txt");
			String temp[]= new String[50];
			String str;
			int j = 0;
			while(!((str = scan.nextLine()).equals("#")))
				temp[j++] = str;
			sort(temp,j);
			for(int k=0;k<j;k++)
				pw.println(temp[k]);
			pw.flush();
			pw.close();
		}
		scan.close();
	}

	public void sort(String s[],int count)
	{
		String temp;
		for(int i=0;i<count;i++)
			for(int j=i+1;j<count;j++)
				if(s[i].compareTo(s[j])>0)
				{
```

```
                        temp = s[i];
                        s[i]=s[j];
                        s[j]=temp;
                }
}

public void mergeFiles()throws IOException
{
        int n = k;
        while(n > 1)
        {
                int count = 1;
                for(int i = 1; i <= n; i+=2)
                {
                        File f1 = new File("list"+i+".txt");
                        File f2 = new File("list"+(i+1)+".txt");
                        File f3 = new File("list" + i + (i+1) + ".txt");
                        BufferedReader br1 = new BufferedReader(new FileReader(f1));
                        BufferedReader br2 = new BufferedReader(new FileReader(f2));
                        PrintWriter pw = new PrintWriter(f3);
                        String name1 = br1.readLine();
                        String name2 = br2.readLine();
                        while(name1 != null && name2 != null)
                        {
                                if(name1.equals(name2))
                                {
                                        pw.println(name1);
                                        pw.println(name2);
                                        name1 = br1.readLine();
                                        name2 = br2.readLine();
                                }

                                else if(name1.compareTo(name2)<0)
                                {
                                        pw.println(name1);
                                        name1 = br1.readLine();
                                }

                                else
                                {
                                        pw.println(name2);
                                        name2 = br2.readLine();
                                }
                        }
                        if(name1 == null)
                        {
```

```
                                        while(name2 != null)
                                        {
                                                pw.println(name2);
                                                name2 = br2.readLine();
                                        }
                                }
                                if(name2 == null)
                                {
                                        while(name1 != null)
                                        {
                                                pw.println(name1);
                                                name1 = br1.readLine();
                                        }
                                }
                                pw.close();
                                br1.close();
                                br2.close();
                                f1.delete();
                                f2.delete();
                                f3.renameTo(new File("list"+count+".txt"));
                                count++;
                        }
                        n/=2;
                }
        }

        public void display()throws IOException
        {
                BufferedReader b = new BufferedReader(new InputStreamReader(new
FileInputStream("list1.txt")));
                String l;
                System.out.println("\nThe merged list is:");
                while((l = b.readLine()) != null)
                        System.out.println(l);
                b.close();
        }
}
```

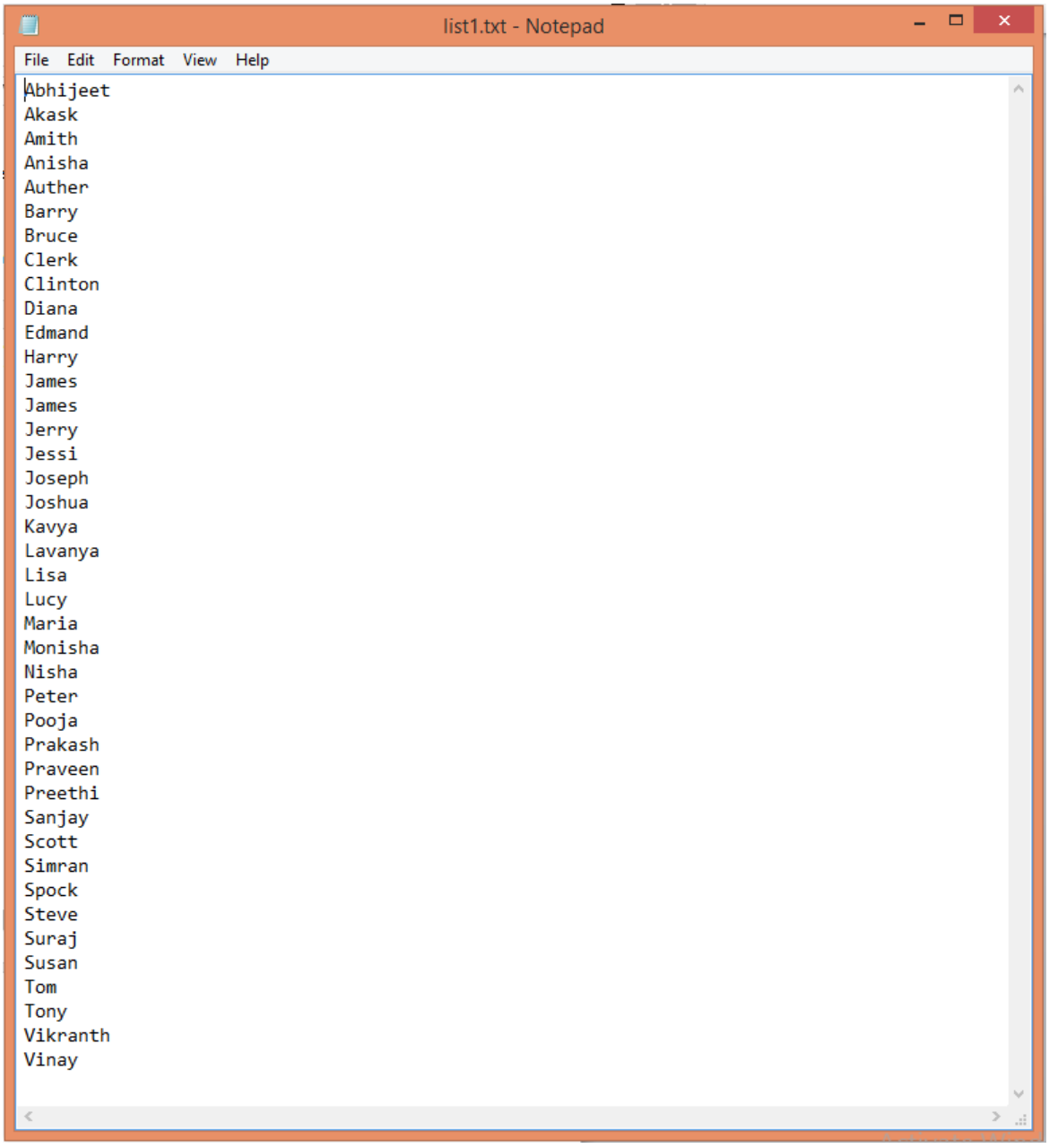
Output:

```
Enter the of names in list 1. Enter # to terminate list
Suraj
Moahn
Abhijeet
Akask
Vikranth
Praveen
#
Enter the of names in list 2. Enter # to terminate list
Prakash
Nisha
Anisha
Sanjay
Joshua
#
Enter the of names in list 3. Enter # to terminate list
Vinay
Amith
Kavya
Lavanya
#
Enter the of names in list 4. Enter # to terminate list
Maria
Joseph
Harry
Simran
#
Enter the of names in list 5. Enter # to terminate list
Pooja
Lisa
Preethi
Jessi
James
#
Enter the of names in list 6. Enter # to terminate list
Tom
Jerry
Scott
James
Spock
#
Enter the of names in list 7. Enter # to terminate list
Steve
Tony
```

Bruce
Clerk
Clinton
#
Enter the of names in list 8. Enter # to terminate list
Auther
Diana
Barry
Peter
Susan
Edmand
Lucy
#

The merged list is:
Abhijeet
Akask
Amith
Anisha
Auther
Barry
Bruce
Clerk
Clinton
Diana
Edmand
Harry
James
James
Jerry
Jessi
Joseph
Joshua
Kavya
Lavanya
Lisa
Lucy
Maria
Moahn
Nisha
Peter
Pooja
Prakash
Praveen
Preethi
Sanjay

Scott
Simran
Spock
Steve
Suraj
Susan
Tom
Tony
Vikranth
Vinay



Merged List