Practical – 5

Aim: Implement Huffman Code to generate binary code when symbol and probabilities are given.

```java
package new_package;


import java.util.*;
class myarraylist{

    ArrayList<Float> freq=new ArrayList<Float>();

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

    ArrayList<String> chars2=new ArrayList<String>();

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

    void addtochars(String a)       {

        chars2.add(a);

    }

    void add(String b,Float a) {

        freq.add(a);

        chars.add(b);

    }

    void setblankcode()    {

        codes.add("");

    }

    void setcode(String ch,String element)        {

        int len=ch.length();

        while(len>0){

            char cha=ch.charAt(len-1);

            int index=-1;

            for(int i=0;i<chars2.size();i++){

                if(chars2.get(i).charAt(0)==cha)
```

```
                index=i;
            }
        System.out.println("index:"+index);
        String temp=codes.get(index);
        codes.set(index, element+temp);
        len--;
        }
    }
    int getindexof(String element)        {
        int index=-1;
        for(int i=0;i<chars2.size();i++){
            if(chars2.get(i).equals(element))
                index=i;
        }
        return index;
    }
    String getchar(int index) {
        return chars.get(index);
    }
    Float getfreq(int index) {
        return freq.get(index);
    }
    void remove(int index) {
        freq.remove(index);
        chars.remove(index);
    }
    void print() {
        System.out.println(freq);
        System.out.println(chars);
```

2161603:Data Compression and Data Retrieval

```java
                System.out.println(codes);

                System.out.println(chars2);

        }

        int extractmin(){

                int index=0;

                for(int i=0;i<freq.size();i++){

                        if(freq.get(index)>freq.get(i)){

                                index=i;

                        }

                }

                System.out.println("Min:"+freq.get(index));

                return index;

        }

}

public class prac5 {

        public static void main(String args[]){

                myarraylist a1=new myarraylist();

                System.out.println("Enter number of characters");

                Scanner sc=new Scanner(System.in);

                int no=sc.nextInt();

                System.out.println("Enter characters and their frequencies");

                for(int i=0;i<no;i++){

                        String s=sc.next();

                        a1.add(s,sc.nextFloat());

                        a1.addtochars(s);

                        a1.setblankcode();

                }

                a1.print();

                while(a1.getfreq(0)!=1){
```

2161603:Data Compression and Data Retrieval

```java
            int i1=a1.extractmin();

            float a=a1.getfreq(i1);

            String s1=a1.getchar(i1);

            System.out.println("i1:"+i1);

            a1.remove(i1);

            int i2=a1.extractmin();

            System.out.println("i2:"+i2);

            float b=a1.getfreq(i2);

            String s2=a1.getchar(i2);

            int t1=a1.getindexof(s1);

            int t2=a1.getindexof(s2);

            System.out.println("t1:"+t1);

            System.out.println("t2:"+t2);

            if(a<=b){

                    a1.setcode(s1, "0");

                    a1.setcode(s2, "1");

            }

            else{

                    a1.setcode(s1, "1");

                    a1.setcode(s2, "0");

            }

            a1.remove(i2);

            a1.add(s1+s2, a+b);

            a1.print();

        }

        sc.close();

    }

}
```

2161603:Data Compression and Data Retrieval

Output:

```
Markers  Properties  Servers  Data Source Explorer  Snippets  Console ✕
<terminated> prac5 [Java Application] C:\Program Files\Java\jdk1.8.0_201\bin\javaw.exe (15-Feb-2019, 9:26:21 pm)
Enter number of characters
5
Enter characters and their frequencies
a 0.4
b 0.2
c 0.2
d 0.1
e 0.1
[0.4, 0.2, 0.2, 0.1, 0.1]
[a, b, c, d, e]

[bcdea]
[11, 00, 01, 100, 101]
[a, b, c, d, e]
```

2161603:Data Compression and Data Retrieval