Program:

```
import java.util.*;

public class productCipher {
    // For Finding the Ascii Value of Characters in a String
    public static int[] ascii(String text){
        int[] arr = new int[text.length()];
        for(int i=0; i<text.length(); i++){
            int ascii = text.charAt(i)-97;
            arr[i]=ascii;
        }
        return arr;
    }

    // Auto Key Encryption Function
    public static char[] autoKeyEn(int[] asciiArr){
        int k=12;
        int[] arr = new int[asciiArr.length];
        arr[0]=(asciiArr[0]+k)%26;

        for(int i=1; i<asciiArr.length; i++){
            arr[i]=(asciiArr[i]+asciiArr[i-1])%26;
        }

        char[] charArr = new char[asciiArr.length];
        for(int i=0; i<charArr.length; i++){
            char ch = (char)(arr[i]+97);
            charArr[i]=ch;
        }

        return charArr;
    }
```

```
    // Auto Key Decryption Function
    public static char[] autoKeyDe(int[] asciiArr, int[]
asciiArr1){
        int k=12;
        int[] arr = new int[asciiArr.length];
        if(asciiArr[0]-k>0){
            arr[0]=(asciiArr[0]-k)%26;
        }
        else{
            arr[0]=(26-(k-asciiArr[0]))%26;
        }

        for(int i=1; i<asciiArr.length; i++){
            if(asciiArr[i]-asciiArr[i-1]>0){
                arr[i]=(asciiArr[i]-asciiArr1[i-1])%26;
            }
            else{
                arr[i]=(26-(asciiArr1[i-1]-asciiArr[i]))%26;
            }
        }

        char[] charArr = new char[asciiArr.length];
        for(int i=0; i<charArr.length; i++){
            char ch = (char)(arr[i]+97);
            charArr[i]=ch;
        }

        return charArr;
    }

    // Rail Fence Encryption Function
```

```
    public static String railFenceEn(String text){
        StringBuilder builder1 = new StringBuilder();
        StringBuilder builder2 = new StringBuilder();

        for(int i=0; i<text.length(); i++){
            if(i%2==0){
                builder1.append(text.charAt(i));
            }
            else{
                builder2.append(text.charAt(i));
            }
        }

        builder1.append(builder2);

        return builder1.toString();
    }

    // Rail Fence Decryption Function
    public static String railFenceDe(String text){
         int len
=(text.length()%2==0)?(text.length()/2):((text.length()/2)+1)
;

         String s1 = text.substring(0, len);
         String s2 = text.substring(len,text.length());
         StringBuilder builder = new StringBuilder();
         int j=0;
         for(int i=0; i<text.length(); i++){
            if(i%2==0){
                builder.append(s1.charAt(j));
            }
```

```
            else{
                builder.append(s2.charAt(j));
                j++;
            }
        }
        return builder.toString();
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the text:\n");
        String text =sc.next();

        // Auto Key Encryption
        int[] textAsciiArr = ascii(text);
        char[] enArr = autoKeyEn(textAsciiArr);
        String enString=new String(enArr) ;
        System.out.println("The Encrypted text using Auto Key
is:");
        System.out.println(enString);

        // Rail Fence Encryption
        String railFenceEnString = railFenceEn(enString);
        System.out.println("The Encrypted text using Rail
Fence is:");
        System.out.println(railFenceEnString);

        // Rail Fence Decryption
        String railFenceDeString =
railFenceDe(railFenceEnString);
        System.out.println("The Decrypted text using Rail
Fence is:");
        System.out.println(railFenceDeString);
```

```
        // Auto Key Decryption
        int[] enTextAscii = ascii(railFenceDeString);
        char[] deArr = autoKeyDe(enTextAscii, textAsciiArr);
        String deString=new String(deArr) ;
        System.out.println("The Decrypted text using Auto Key
is:");
        System.out.println(deString);
    }
}
```

OUTPUT:

```
Enter the text:
association
The Encrypted text using Auto Key is:
mskgqkitbwb
The Encrypted text using Rail Fence is:
mkqibbsgktw
The Decrypted text using Rail Fence is:
mskgqkitbwb
The Decrypted text using Auto Key is:
association
PS C:\Users\HP\OneDrive\Desktop\JAVA>
```