interface LCD {  
 void write(String[] s);  
 void read(String[] s);  
}  
  
class Core implements LCD {  
 BufferedReader in = new BufferedReader(new InputStreamReader(System.in));  
  
 public void write(String[] s) {  
 System.out.print("INPUT: ");  
 try {  
 s[0] = in.readLine();  
 } catch (IOException ex) {  
 ex.printStackTrace();  
 }  
 }  
  
 public void read(String[] s) {  
 System.out.println("Output: " + s[0]);  
 }  
}  
  
class Decorator implements LCD {  
 private LCD inner;  
  
 public Decorator(LCD i) {  
 inner = i;  
 }  
  
 public void write(String[] s) {  
 inner.write(s);  
 }  
  
 public void read(String[] s) {  
 inner.read(s);  
 }  
}  
  
class Scramble extends Decorator {  
 public Scramble(LCD inner) {  
 super(inner);  
 }  
  
 public void write( String[] s ) {  
 super.write(s);  
 System.out.println("encrypt:");  
 StringBuilder sb = new StringBuilder(s[0]);  
 for (int i=0; i < sb.length(); i++) {  
 sb.setCharAt(i, (char)(sb.charAt(i) - 5));  
 }  
 s[0] = sb.toString();  
 }  
  
 public void read(String[] s) {  
 StringBuilder sb = new StringBuilder(s[0]);  
 for (int i=0; i < sb.length(); i++) {  
 sb.setCharAt(i, (char)(sb.charAt(i) + 5));  
 }  
 s[0] = sb.toString();  
 System.out.println( "decrypt:" );  
 super.read(s);  
 }  
}  
  
public class DecoratorDemo extends Decorator {  
 BufferedReader in = new BufferedReader(new InputStreamReader(System.in));  
  
 public DecoratorDemo(LCD inner) {  
 super(inner);  
 }  
  
 public void write(String[] s) {  
 System.out.print("PASSWORD: ");  
 try {  
 in.readLine();  
 } catch (IOException ex) {  
 ex.printStackTrace();  
 }  
 super.write( s );  
 }  
  
 public void read(String[] s) {  
 System.out.print("PASSWORD: ");  
 try {  
 in.readLine();  
 } catch (IOException ex) {  
 ex.printStackTrace();  
 }  
 super.read(s);  
 }  
  
 public static void main(String[] args) {  
 LCD stream = new DecoratorDemo(new Scramble(new Core()));  
 String[] str = {""};  
 stream.write(str);  
 System.out.println("main: " + str[0]);  
 stream.read(str);  
 }  
}