

牛客网-华为机试练习题 28

题目描述

1、对输入的字符串进行加解密，并输出。

2加密方法为：

当内容是英文字母时则用该英文字母的后一个字母替换，同时字母变换大小写，如字母a时则替换为B；字母z时则替换为a；

当内容是数字时则把该数字加1，如0替换1，1替换2，9替换0；

其他字符不做变化。

3、解密方法为加密的逆过程。

接口描述：

实现接口，每个接口实现1个基本操作：

`void Encrypt (char aucPassword[], char aucResult[])`：在该函数中实现字符串加密并输出

说明：

1、字符串以\0结尾。

2、字符串最长100个字符。

`int unEncrypt (char result[], char password[])`：在该函数中实现字符串解密并输出

说明：

1、字符串以\0结尾。

2、字符串最长100个字符。

输入描述:

输入说明

输入一串要加密的密码

输入一串加过密的密码

输出描述:

输出说明

输出加密后的字符

输出解密后的字符

示例1

输入

abcdefg

BCDEFGH

输出

BCDEFGH

abcdefg

解决代码

```
import java.util.Scanner;

public class Main{
    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);
        String str1 = sc.nextLine();
        String str2 = sc.nextLine();
        for(int i=0;i<str1.length();i++){
            System.out.print(solution_encode(str1.charAt(i)));
        }
        System.out.print('\n');

        for(int i=0;i<str2.length();i++){
            System.out.print(solution_decode(str2.charAt(i)));
        }

        }

    public static char solution_encode(char ch){
        if(ch>='0' && ch<='8'){
            return (char)(ch+1);
        }else if(ch=='9'){
            return '0';
        }if(ch>='A' && ch<='Z'){
            return (char)(ch+33);
        }
        if(ch=='Z'){
            return 'a';
        }
        if(ch>='a' && ch<='z'){
            return (char)(ch-32+1);
        }
        if(ch=='z'){
            return 'A';
        }
        return ch;
    }

    public static char solution_decode(char ch){
        if(ch=='0'){
            return '9';
        }
        if(ch>'0' && ch<='9'){
            return (char)(ch-1);
        }
        if(ch=='a'){
            return 'z';
        }
        if(ch>'a' && ch<='z'){
            return (char)(ch-32-1);
        }
        if(ch=='A'){
            return 'z';
        }
        if(ch>'A' && ch<='Z'){
            return (char)(ch+32-1);
        }
    }
}
```

```

    }
    return ch;
}
}

```

```

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

```

```

public class Main{
    public static void main(String[] args) throws IOException{
        BufferedReader bf = new BufferedReader(new InputStreamReader(System.in));
        String str = "";
        while((str=bf.readLine())!=null){
            String str1 = Encrypt(str);
            System.out.println(str1);
            str=bf.readLine();
            String str2=unEncrypt(str);
            System.out.println(str2);
        }
    }
}

```

```

private static String Encrypt(String line){
    char[] cha = line.toCharArray();
    StringBuilder sb= new StringBuilder();
    for(int i=0;i<cha.length;i++){
        if (cha[i]>='a'&&cha[i]<='z'){
            if(cha[i]=='z'){
                sb.append('A');
            }else{
                sb.append((char)(cha[i]+1-32));
            }
        }else if(cha[i]>='A'&&cha[i]<='Z'){
            if(cha[i]=='Z'){
                sb.append('a');
            }else{
                sb.append((char)(cha[i]+32+1));
            }
        }else if(cha[i]>='0'&&cha[i]<='9'){
            if(cha[i]=='9'){
                sb.append('0');
            }else{
                sb.append(cha[i]-48+1);
                // sb.append((char)(cha[i]+1));
            }
        }else{
            sb.append(cha[i]);
        }
    }
    return sb.toString();
}
}

```

```

private static String unEncrypt(String line){
    char[] cha = line.toCharArray();
    StringBuilder sb= new StringBuilder();
    for(int i=0;i<cha.length;i++){
        if (cha[i]>='a'&&cha[i]<='z'){
            if(cha[i]=='a'){
                sb.append('Z');
            }else{

```

```
        sb.append((char)(cha[i]-1-32));
    }
} else if(cha[i]>='A'&&cha[i]<='Z'){
    if(cha[i]=='A'){
        sb.append('z');
    } else{
        sb.append((char)(cha[i]+32-1));
    }
} else if(cha[i]>='0'&&cha[i]<='9'){
    if(cha[i]=='0'){
        sb.append('9');
    } else{
        sb.append(cha[i]-48-1);
        // System.out.println((char)(cha[i]-1));
        //System.out.println(cha[i]-48-1);
    }
} else{
    sb.append(cha[i]);
}
}
return sb.toString();
}
```