

时间限制：C/C++ 1秒，其他语言2秒 空间限制：C/C++ 32M，其他语言64M 热度指数：1105

■ 算法知识视频讲解

```
 1  2  3
   ABC DEF
 4  5  6
 GHI JKL MNO
 7  8  9
 PQRS TUV WXYZ
   0
```

上图是一个电话的九宫格，如你所见一个数字对应一些字母，因此在国外企业喜欢把电话号码设计成与自己公司名字相对应。例如公司的Help Desk号码是4357，因为4对应H、3对应E、5对应L、7对应P，因此4357就是HELP。同理，TUT-GLOP就代表888-4567、310-GINO代表310-4466。  
NowCoder刚进入外企，并不习惯这样的命名方式，现在给你一串电话号码列表，请你帮他转换成数字形式的号码，并去除重复的部分。

**输入描述:**

输入包含多组数据。

每组数据第一行包含一个正整数 $n$  ( $1 \leq n \leq 1024$ )。

紧接着 $n$ 行，每行包含一个电话号码，电话号码仅由连字符“-”、数字和大写字母组成。  
没有连续出现的连字符，并且排除连字符后长度始终为7（美国电话号码只有7位）。

**输出描述:**

对应每一组输入，按照字典顺序输出不重复的标准数字形式电话号码，即“xxx-xxxx”形式。

每个电话号码占一行，每组数据之后输出一个空行作为间隔符。

**示例1**

输入

```
12
4873279
ITS-EASY
888-4567
3-10-10-10
888-GLOP
TUT-GLOP
967-11-11
310-GINO
F101010
888-1200
-4-8-7-3-2-7-9-
487-3279
4
UTT-HELP
TUI-GLOP
310-GINO
000-1213
```

输出

```
310-1010
310-4466
487-3279
888-1200
888-4567
967-1111

000-1213
310-4466
888-4357
888-4567
```

```

1 #include<iostream>
2 #include<string>
3 #include<set>
4 using namespace std;
5
6 int main()
7 {
8     int n;
9     string NUM = "2223334445556667778889999";
10    while (cin>>n)
11    {
12        set<string> s;
13        for (int i=0;i<n;i++)
14        {
15            string res="";
16            string str;
17            int cnt=0;
18            cin>>str;
19            for (int j=0;j<str.size();j++)
20            {
21                if (str[j]>='A'&&str[j]<='Z')
22                {
23                    res+=NUM[str[j]-'A'];
24                    cnt++;
25                    if (cnt==3)
26                    {
27                        res+=" ";
28                    }
29                }
30                else if (str[j]>='0'&&str[j]<='9')
31                {
32                    res+=str[j];
33                    cnt++;
34                    if (cnt==3)
35                    {
36                        res+=" ";
37                    }
38                }
39            }
40            s.insert(res);
41        }
42        for (set<string>::iterator it=s.begin();it!=s.end();it++)
43        {
44            cout<<*it<<endl;
45        }
46        cout<<endl;
47    }
48    return 0;
49 }

```