

国际摩尔斯密码定义一种标准编码方式，将每个字母对应于一个由一系列点和短线组成的字符串，比如: "a" 对应 ".-.", "b" 对应 "-...", "c" 对应 "-.-.", 等等。

为了方便，所有26个英文字母对应摩尔斯密码表如下：

```
[".-.", "-...", "-.-.", "-..", ".", ".-.", "--.", "...", "..", ".---", "-.-", ".-..", "-.", ".---", ".--.", "--.-", ".-.", "...", "-", ".-.", "...-", ".-.-", "-.-.", "-.-.", "-..."]
```

给定一个单词列表，每个单词可以写成每个字母对应摩尔斯密码的组合。例如，"cab" 可以写成 "-.-.-...-", (即 "-.-." + "-..." + "-."字符串的结合)。我们将这样一个连接过程称作单词翻译。

返回我们可以获得所有词不同单词翻译的数量。

例如：

输入: words = ["gin", "zen", "gig", "msg"]

输出: 2

解释：

各单词翻译如下：

```
"gin" -> "--...-."
"zen" -> "--...-."
"gig" -> "--...--."
"msg" -> "--...--."
```

共有 2 种不同翻译， "--...-." 和 "--...--."。

注意：

- 单词列表words 的长度不会超过 100。
- 每个单词 words[i] 的长度范围为 [1, 12]。
- 每个单词 words[i] 只包含小写字母。

分析，直接按照字符去解就ok了，因为现在有map，所以已经存在的，用map进行标记就可以进行判断了，c++中map值可以按照变量初始化，bool初值为false,所以进行取反判断即可。用num动态记录有几个单词。

代码

```
class Solution {
public:
    int uniqueMorseRepresentations(vector<string>& words) {
        map<string, bool> bj;
```

```

        string code[]={".-","-...","-.-","-..",".", "-.-.", "-.", "-...", "..", "-
--", "-.-", "-..", "--", "-.", "----", "-.-.", "-.-.", "-.", "-...", "-", "-.-", "-.-.", "-.-
", "-.-.", "-.-.", "-.-."};
        int num=0;
        for(int i=0;i<words.size();i++){
            string s="";
            for(int j=0;j<words[i].size();j++){
                s+=code[words[i][j]-'a'];
            }
            if(!bj[s]){
                bj[s]=true;
                num++;
            }
        }
        return num;
    }
};

```