

## P1603 斯诺登的密码

## 题目描述

2013 年 X 月 X 日，俄罗斯办理了斯诺登的护照，于是他混迹于一架开往委内瑞拉的飞机。但是，这件事情太不周密了，因为FBI的间谍早已获悉他的具体位置——但这不是最重要的——最重要的是如果要去委内瑞拉，那么就要经过古巴，而经过古巴的路在美国的掌控之中。

丧心病狂的奥巴马迫降斯诺登的飞机，搜查时却发现，斯诺登杳无踪迹。但是，在据说是斯诺登的座位上，发现了一张纸条。纸条由纯英文构成：Obama is a two five zero.（以 . 结束输出，只有 6 个单词+一个句号，句子开头如没有大写亦为合法）这句话虽然有点无厘头，但是警官陈珏骛发现这是一条极其重要的线索。他在斯诺登截获的一台笔记本中找到了一个 C++ 程序，输入这条句子后立马给出了相对应的密码。陈珏骛高兴得晕了过去，身为警官的你把字条和程序带上了飞机，准备飞往曼哈顿国际机场，但是在飞机上检查的时候发现——程序被粉碎了！飞机抵达华盛顿只剩5分钟，你必须在这 5 分钟内编写（杜撰）一个程序，免受上司的 10000000000%10 大板。破译密码的步骤如下：

(1) 找出句子中所有用英文表示的数字 ( $\leq 20$ ) ( $\leq 20$ )，列举在下：

正规：one two three four five six seven eight nine ten eleven twelve thirteen  
fourteen fifteen sixteen seventeen eighteen nineteen twenty

非正规：a both another first second third。为避免造成歧义，another 算作 11 处理。

(2) 将这些数字平方后对 100100 取模，如 00,05,11,19,86,9900,05,11,19,86,99。

(3) 把这些两位数按数位排成一行，组成一个新数，如果开头为 00，就去 00。

(4) 找出所有排列方法中最小的一个数，即为密码。

// 数据已经修正 By absi2011 如果还有问题请联系我

## 输入格式

一个含有 6 个单词的句子。

## 输出格式

一个整型变量（密码）。如果没有符合要求的数字出现，则输出 0。

## 输入样例

Black Obama is two five zero .

## 输出样例

425

## 解析

该题比较令人费解的是要求排序后的最小值。根据题目我们可以猜测出，题目实按照实际数字的大小进行排序，而不是组成的结果。否则，题目的最终结果应该输出254，而不是425了。

## 编码

```
#include<bits/stdc++.h>

using namespace std;
//将英文单词进行定义
string def[27] = {"zero", "one", "two", "three", "four", "five",
                  "six", "seven", "eight", "nine", "ten", "eleven",
                  "twelve", "thirteen", "fourteen", "fifteen",
                  "sixteen", "seventeen", "eighteen", "nineteen",
                  "twenty", "a", "both", "another", "first",
                  "second", "third"};
//英文字母的平方值
int di[27] = { 0, 1, 4, 9, 16, 25,
               36, 49, 64, 81, 00, 21,
               44, 69, 96, 25, 56, 89,
               24, 61, 0, 1, 4, 1,
               1, 4, 9};
unsigned long long int a[10], top;

bool beFirst = true;
int i, j;
string input;

int main() {
    //总共6个单词
    for (i = 1; i <= 6; i++) {
        cin >> input;
        //遍历全部的单词定义，观察当前词组是否为数字
        for (j = 0; j <= 26; j++) {
            //strcmp(s1,s2);如果他们相同，返回0
            if (!input.compare(def[j])) {
                //用数组存储
                a[++top] = di[j];
                break; //立即停止寻找
            }
        }
    }
    //贪心，使越小的数越靠前输出
    //调用系统默认的排序函数，将数组内容按照升序进行排列
```

```
sort(a + 1, a + top + 1);
for (i = 1; i <= top; i++) {
    if (!beFirst)//如果不是第一位
    {
        printf("%02d", a[i]); //限制格式输出
    } else {
        if (a[i]) {
            printf("%d", a[i]);
            beFirst = false;
        }
    }
}
//没有找到合法数字
if (beFirst) {
    printf("0"); //特判
}
return 0;
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

