第一讲：为什么计算机能解题?

姓名：丁保荣　　**学号**：171860509

2017 年 10 月 18 日

请独立完成作业，不得抄袭。
若参考了其它资料，请给出引用。
鼓励讨论，但需独立书写解题过程。

第一部分　作业

题目 (UD: 1.2)

Find a word (written in standard capital letters) that reads the same forward and backward and is still the same forward and backward when rotated around its center 180◦. Your solution needs to appear in a standard dictionary of some language. （抄写或简述题目）

注:

解答:

根据题目要求可知单词成中心对称且回文，所以组成单词的字母得本身成中心对称，或与其他字母成中心对称。所以符合条件的字母有：1. 本身成中心对称系列：H,I,Z,O,S,N 2. 与其他字母成中心对称：M 和 W. 所以将上述字母组合，可得到诸如 SOS,NOON, 以及 WOW 和 MOM 等。

题目 (UD: 1.3)

Solve the following anagrams. The first three are places (in the ge- ographical sense), and the fourth is a place in which you might live. All can be rearranged to form a single word. (a) NOVACURVE; (b) NINESLAPNAVY; (c) IHELDAHIPPAL; (d) DIRTYROOM. Note:You may have to find out exactly what ananagram is.This is partofPo´lya's first point on the list.

解答:

answers 1.vancourve 2.pennsylvania 3.philadelphia 4.dormitory

sourcecode:

```
#include <iostream>
```

```
using namespace std;
struct stringarray
{
    string word;
};

int compare(string word,int arr[]);

int main(void)
{
    string anagram;
    int sum=0; //单词长度
    stringarray dictionary[10000];
    cout <<"Please enter an anagram"<<endl;
    getline(cin,anagram);
    int num=0; //字典单词数目
    int arr[26]={0};
    for(int i=0;i<=anagram.size()-1;i++)
    {
        if ((anagram[i]<=122)&&anagram[i]>=97)
        {
            sum++;
            switch(anagram[i])
            {
                case 97:arr[0]++; break;
                case 98:arr[1]++; break;
                case 99:arr[2]++; break;
                case 100:arr[3]++; break;
                case 101:arr[4]++; break;
                case 102:arr[5]++; break;
                case 103:arr[6]++; break;
                case 104:arr[7]++; break;
                case 105:arr[8]++; break;
                case 106:arr[9]++; break;
                case 107:arr[10]++; break;
                case 108:arr[11]++; break;
                case 109:arr[12]++; break;
                case 110:arr[13]++; break;
                case 111:arr[14]++; break;
                case 112:arr[15]++; break;
```

```
                case 113:arr[16]++; break;
                case 114:arr[17]++; break;
                case 115:arr[18]++; break;
                case 116:arr[19]++; break;
                case 117:arr[20]++; break;
                case 118:arr[21]++; break;
                case 119:arr[22]++; break;
                case 120:arr[23]++; break;
                case 121:arr[24]++; break;
                case 122:arr[25]++; break;
                default: break;

            }
        }
    }
    cout <<"Please enter the sum of the words in the dictionary"<<endl;
    cin >>num;
    cout<<"Please enter the words in the dictionary,each line only one word"<<endl;
    cin.get();
    for(int i=1;i<=num;i++)
        getline(cin,dictionary[i].word);
    for(int i=1;i<=num;i++)
        if (dictionary[i].word.size()==sum)
            if (compare(dictionary[i].word,arr)==0)
            {
                cout<<"the place is "<<dictionary[i].word;
                break;
            }
    return 0;
}




int compare(string word,int arr[])
{
    int arr2[26]={0};
    for(int i=0;i<=word.size()-1;i++)
    {
        if ((word[i]<=122)&&word[i]>=97)
```

```
    {
        switch(word[i])
        {
            case 97:arr2[0]++; break;
            case 98:arr2[1]++; break;
            case 99:arr2[2]++; break;
            case 100:arr2[3]++; break;
            case 101:arr2[4]++; break;
            case 102:arr2[5]++; break;
            case 103:arr2[6]++; break;
            case 104:arr2[7]++; break;
            case 105:arr2[8]++; break;
            case 106:arr2[9]++; break;
            case 107:arr2[10]++; break;
            case 108:arr2[11]++; break;
            case 109:arr2[12]++; break;
            case 110:arr2[13]++; break;
            case 111:arr2[14]++; break;
            case 112:arr2[15]++; break;
            case 113:arr2[16]++; break;
            case 114:arr2[17]++; break;
            case 115:arr2[18]++; break;
            case 116:arr2[19]++; break;
            case 117:arr2[20]++; break;
            case 118:arr2[21]++; break;
            case 119:arr2[22]++; break;
            case 120:arr2[23]++; break;
            case 121:arr2[24]++; break;
            case 122:arr2[25]++; break;
            default: break;

        }
    }
}
for(int i=0;i<=25;i++)
{
    if (arr[i]!=arr2[i])
    {
        return 1;
    }
```

```
    }
    return 0;

}
```

题目 (UD: 1.4)

Suppose n teams play in a single game elimination tournament. How many games are played? An example of such tournaments are the various categories of the U. S. Open tennis tournament; for example, women's singles. Note: Pay special attention to the first entry of Po´lya's list: "Is it possible to satisfy the condition?"（抄写或简述题目）

注:

解答:

设总场数为 x

1.当n = 2^t 时，x=1+2^1+2^2+……+2^(t-1)=2^t-1

2.当2^t<n<2^(t+1)时，x=1+2^1+……+2^(t-1)+n-2^t=2^t-1+n-2^t

（第一场时有部分队伍轮空，只有(n-2^t) *2支队伍打，

淘汰完(n-2^t)支队伍后，便与上一种情况一样了）

所以两种情况可以统一起来：x=2^t-1+n-2^t

题目 (UD: 1.5)

Suppose you are all alone in a strange house. There are seven identical closed doors. The bathroom is behind exactly one of them. Is it more likely, less likely, or equally likely that you find the bathroom on the first try than on the third try? Why? （抄写或简述题目）

注:

证明:

It is equally likely that you find the bathroom on the first try than on the third try.

1.Let us consider "the first try "first, there are seven closed doors while there is only one behind which a bathroom lies, so p(succeed on the first try)=1/7

2.Then let's consider "the third try",it means that you fail on the first and second try until you succeed on the third try , so p(succeed on the third try)=(6/7)*(5/6)*(1/5)=1/7.

So it is equally likely that you find the bathroom on the first try than on the third try. □

题目 (UD: 1.6)

The following message is encoded using a shifted alphabet just as in Exercise 1.1. (Of course, the shift number n is not the same as in the exercise!) What does the message say? RDSXCVIWTDGNXHUJCLTLXAAATPGCBDGTPQDJIXIAPITG(抄写或简述题目)

注:

解答:

answer:CODINGTHEORYISFUNWEWILLLEARNMOREABOUTITLATER n=11

source code:

```
#include <iostream>
#include<string>
int main(void) {
    using namespace std;
    string code;
    cout << "Enter the code";
    getline(cin,code);
    cout <<endl;
    int length=code.size();
    for(int j=1;j<=26;j++)
    {
        for(int i=0;i<length;i++)
                if (code[i]>=65 && code[i]<=90)
                {
                    if ((code[i]+1)<=90)
                        code[i]=code[i]+1;
                    else
                        code[i]=code[i]-26+1;
                }
        cout << code <<j <<endl;
    }
    return 0;
}
```

题目 (UD: 1.7)

Give a detailed description of all points in three-space that are equidis- tant from the

x-axis and the yz-plane. Once you decide on the answer, write the solution up carefully. Pay particular attention to your notation. （抄写或简述题目）

注:

解答:

设A(x,y,z)为满足题意的点， 并取B(x,0,0) C(0,y,z)
所以AB垂直于x轴，所以A到x轴的距离为sqrt(y^2+z^2)
所以AC垂直于yz平面，所以A到yz平面的距离为sqrt(x^2)
因为A到x轴的距离等于A到yz平面的距离(由题意得)，
所以AB=AC，所以sqrt(y^2+z^2)=sqrt(x^2)，所以x^2=y^2+z^2
所以综上A点坐标应满足的条件是x^2=y^2+z^2

题目 (UD: 1.8)

The following is a classic problem in mathematics. Though there are many variations of this problem, the standard one is the following. You are given 12 coins that appear to be identical. However, one of the coins is counterfeit, and the weight of this coin is slightly different than that of the other 11. Using only a two-pan balance, what is the smallest number of weighings you would need to find the counterfeit coin? (Think about a simpler, similar problem.) (See I. Peterson's website [82] for a discussion of this problem.)（抄写或简述题目）

注: 一定是错误的解答，想用局部最优解，但无法得出总体最优解

解答:

1. 先考虑第一次操作能把范围缩减的最小值：放在天平上的个数应该是相同的才好比较

1.1 两边各两个：无意义

1.2 两边各三个：则在天平上的共有六个（设为集合 A），不在天平上的也有六个（设为集合 B），此时，若天平平衡，则假币在 B 集合中；若天平不平衡，则假币在 A 集合中；所以此时将范围缩小到了六个；

1.3 两边各四个：则在天平上的共有八个（设为集合 A），不在天平上的有四个（设为集合 B）此时，若天平平衡，则假币在 B 集合中；若天平不平衡，则假币在 A 集合中；因为要考虑所有情况，所以以最糟糕的情况的范围为准，所以此时将范围缩小到了八个；

1.4 两边各五个：此时将范围缩小到了十个；

所以经过一次操作的最优范围是六个；

2.1 两边各三个：无意义

2.2 两边各两个：则在天平上的共有四个（设为集合 A），不在天平上的有两个（设为集合 B）此时，若天平平衡，则假币在 B 集合中；若天平不平衡，则假币在 A 集合中；所以此时将范围缩小到了四个所以经过两次操作的最优范围是四个；

3 在天平的一端放入已经鉴别好的真币两枚，另一端放入未鉴别硬币两枚（设为集合 A），还有两枚未鉴别硬币两枚不在天平上（设为集合 B）此时，若天平平衡，则假币在集合 B 中，若天平不平衡，则假币在集合 A 中；所以此时将范围缩小到了两个；第四次操作就可以把假币区分出来，不再赘述。

第二部分　订正

题目 (题号)
题目。

错因分析：　简述错误原因（可选）。

订正：
正确解答。

第三部分　反馈

你可以写：

- 对课程及教师的建议与意见
- 教材中不理解的内容
- 希望深入了解的内容
- 等