ACM/ICPC 校内选拔赛

姓名：　　　　　　　　　　学号：

时间：14：30 – 17：30，3 个小时

答题说明：每道题请给出解题的思路，算法的描述，用到的公式，并且给出伪代码，伪代码只要给出核心部分，输入输出可以不写。

1. 写一个函数，把一个句子倒置．使得句子中的单词,标点和空格的顺序颠倒，但单词中字母的顺序不变．该句子只可能包含字母，空格，逗号和句号．

```
void sentence_reverse( char* sentence );
```

比如　　sentence = "Hi, new ustc icpcers."

倒置之后 sentence 为 ".icpcers ustc new ,Hi"

要求时间复杂度不超过 O(n)，内存的使用越小越好

参考答案：

```
//首先将该句子倒置
//找到每个单词
//将单词再次倒置
void strrev( char *start, char *end )
{
    char temp;
    while( start < --end )
    {
            temp = *start;
            *start = *end;
            *end = temp;
            ++start;
        }
    }

    void sentence_reverse( char *str )
    {
        char *sen_end = str, *start, *end;
        while( *sen_end )
            ++sen_end;

        strrev( str, sen_end );
```

```
    start = str;

    while( start != sen_end )
    {
        while( *start == ' ' || *start == ',' || *start == '.' )
            ++start;

        end = start;
        while( *end >= 'a' && *end <= 'z' || *end >= 'A' && *end<= 'Z' )
            ++end;

        strrev( start, end );

        start = end;
    }
}
```

2. 你打算去参加一个射击游戏。第一发子弹命中目标的概率是 50%。接下来第 n（n>1）发子弹命中的概率按如下规则变化：如果上一发子弹命中目标，那么此发命中的概率为 40%,如果上一发没有命中，那么此发命中的概率为 60%。总共命中目标 4 次才会有奖品。给出获胜所需要子弹数的期望值。

E{X} = (1*.5+2*.5*.6+3*.5*.4*.6+4*.5*.4^2*.6+5*.5*.4^3*.6+...) +
3*(1*.4+2*.6*.6+3*.6*.4*.6+4*.6*.4^2*.6+5*.6*.4^3*.6+...)
= 77/6

3. Joseph 问题是一个非常著名的问题. 有 n 个人排成一个圈，分别编号为 1，2，……，n。编号为 1 的人开始从 1 报数，报到 m 倍数的人会被剔除出队，直到队伍中剩下一个人。例如：当 n = 6，m = 5，剔除出队的人的编号依次为：5,4,6,2,3。最后剩下编号为 1 的人。

你的任务是给出当 m=2 时，有多少个 n（1 <= n <= 2005）满足最后剩下的人的编号为 1。

当 n = 2^m, m = 0,1,2,3,...时满足要求。[1,2005]共有 11 个

4. 给出 n 个数对(A_i, B_i) (1<=i<=n)，其中 A_i<=A_{i+1} (1<=i<n)。现在要选择一些数对使得这些数对的第二个分量的和尽量大而且任意两个选出的数对的第一个分量的差不超过给出的数 m。

请给出一个解决这个问题的时间复杂度不高于 O(n)的算法。

第四题用动态规划解决。
状态是 f(i)，从前 i 个数对选出的最大和。
转移是 f(i)=max{f(g(i))+Bi,f(i-1)}，讨论第 i 个数对是否选中 ，
其中 g(i)=min{j|1<=j<i 且 Ai-Aj<=m}。
直接做是平方的，需要一点小技巧找 g(i)。

容易看出 $g(i)$ 是 i 的单调不减函数，所以在动态规划计算状态 $f(i)$ 的时候，
从 $g(i-1)$ 更新到 $g(i)$ 只需要向后扫描而不需要回溯，所以时间复杂度是线性的。

5. 定义 $f(0)=1$，$f(n)$等于将 n 表示成一些 2 的幂之和且每种幂最多只能用两次的方法数。比如，$f(10)=5$，因为有 5 种不同的方法来表示 10：1+1+8, 1+1+4+4, 1+1+2+2+4, 2+4+4 和 2+8. 请设计一个时间复杂度尽量低的算法来计算 $f(n)$。

解答：讨论 1 的取的个数容易得到递推式：
$f(2n)=f(n-1)+f(n)$
$f(2n+1)=f(n)$
按如下矩阵乘法公式同时递推计算 $f(n)$ 和 $f(n-1)$。

$$\left|\begin{matrix} f(2*n+1) \\ \\ f(2*n) \end{matrix}\right| = \left|\begin{matrix} 1 & 0 \\ & \\ 1 & 1 \end{matrix}\right| \left|\begin{matrix} f(n) \\ \\ f(n-1) \end{matrix}\right|$$

$$\left|\begin{matrix} f(2*n) \\ \\ f(2*n-1) \end{matrix}\right| = \left|\begin{matrix} 1 & 1 \\ & \\ 0 & 1 \end{matrix}\right| \left|\begin{matrix} f(n) \\ \\ f(n-1) \end{matrix}\right|$$

时间复杂度为 $O(\lg(n))$。