t -> 0
(((a^t+b^t)/2)^(1/t)-(ab)^.5)/t
=(ab)^(1/2)/4((lna)^2+(lnb)^2)

错位相减法

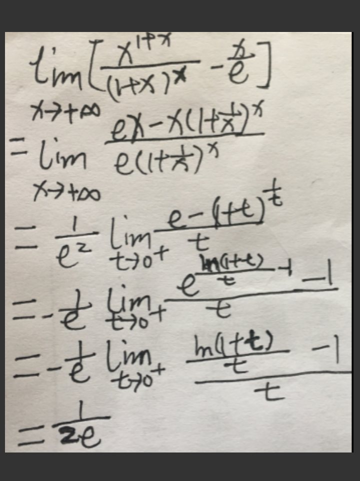
形如An=BnCn，其中{Bn}为等差数列，{Cn}为等比数列；

分别列出Sn，再把所有式子同时乘以等比数列的公比q，即q·Sn；然后错开一位，两个式子相减。

应用于等比数列与等差数列相乘的形式。

4个相同红球和2个相同黑球有多少种排列顺序？ 答案：C(2,6)

∫(secx)^2dx
=x+∫(tanx)^2dx #分步积分发，令u=sinx
=tanx+C



题目： http://poj.org/problem?id=2229

题意： 给定一个正整数n，求有多少种方法把它写成若干个2幂次的和。

分析： 本题可以用递推的思路，我们可以明确，对于一个正整数n，它的2的幂次和的表示分情况讨论

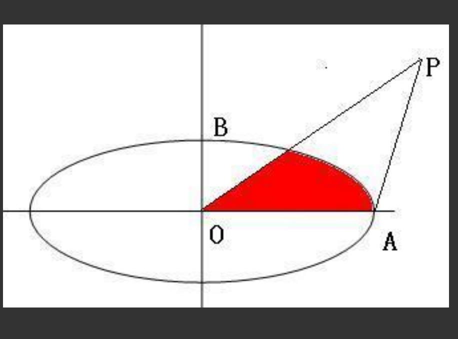
（1）如果n为奇数，那么在这个表示中一定含有一个1，把这个1减去，就是$n-1$的情况了。
（2）如果n为偶数，那么也分情况，得到$dp[i] = dp[i-2] + dp[i >> 1]$。

代码：

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
const int MOD = 1000000000;
const int N = 1000005;

int dp[N];

void Init()
{
    dp[1] = 1;
    dp[2] = 2;
    for(int i = 3; i < N; i++)
    {
        if(i & 1) dp[i] = dp[i - 1] % MOD;
        else      dp[i] = (dp[i - 2] + dp[i >> 1]) % MOD;
    }
}

int main()
{
    int n;
    Init();
    while(cin>>n)
        cout<<dp[n]<<endl;
    return 0;
}
```

题目： http://acm.hdu.edu.cn/showproblem.php?pid=1714

题意： 如下图，给定椭圆的标准方程和点P的坐标，求红色区域的面积。



分析： 首先求直线OP与椭圆的交点坐标(x_1, y_1)，然后得到面积公式为

$$S = \frac{1}{2}x_1y_1 + \frac{b}{a}\int_{x_1}^{a}\sqrt{a^2-x^2}dx$$

根据如下公式

$$\int\sqrt{a^2-x^2}dx = \frac{x}{2}\sqrt{a^2-x^2} + \frac{a^2}{2}\arcsin\frac{x}{a} + C$$

化简后最终结果为

$$\begin{aligned} S &= \frac{ab}{2}\left(\arcsin 1 - \arcsin\frac{x_1}{a}\right) \\ &= \frac{ab}{2}\left(\frac{\pi}{2} - \arcsin\frac{x_1}{a}\right) \\ &= \frac{ab}{2}\arccos\frac{x_1}{a} \end{aligned}$$

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <iomanip>
#include <cmath>

using namespace std;

int main()
{
    int T;
    cin>>T;
    while(T--)
    {
        double a, b, x, y;
        cin>>a>>b>>x>>y;
        if(y < 0) y = -y;
        double x1 = a * b * x / sqrt((a * a * y * y + b * b * x * x));
        x1 /= a;
        double t = acos(x1);
        double area = t * b * a / 2;
        cout<<fixed<<setprecision(2)<<area<<endl;
    }
    return 0;
}
```

题目： http://acm.hdu.edu.cn/showproblem.php?pid=2554

题意： 给定n对数，每对数的值分别为$1, 2, 3, ..., n$，现在要求判断是否存在一个长度为$2n$的序列，使得对所有的数，值为i的两个相同数之间所夹的数的个数为i。

分析： 假设数i的第一个位置为a_i，第二个位置为b_i，那么有$b_i - a_i = i + 1$，继续得到

$$\sum_{i=1}^{n}(b_i - a_i) = \frac{n(n+3)}{2}$$

而对于它们本身会有

$$\sum_{i=1}^{n}(b_i + a_i) = n(2n+1)$$

这样解得

$$\sum_{i=1}^{n}a_i = \frac{n(3n-1)}{4}$$
$$\sum_{i=1}^{n}b_i = \frac{5n(n+1)}{2}$$

接下来，只要保证它们是整数就行。

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
typedef long long LL;

int main()
{
    int n;
    while(cin>>n && n)
    {
        if(n % 4 == 0 || (3 * n - 1) % 4 == 0) puts("Y");
        else puts("N");
    }
    return 0;
}
```

题目： http://acm.hdu.edu.cn/showproblem.php?pid=1214

题意： 对于一个有环序列，每次只能对调相邻两位数字，要得到逆序至少需要操作多少次？

分析： 在直线上移动很简单，即冒泡排序的方法，所以最少移动次数为

$$num = \frac{n(n-1)}{2}$$

那么，在圆环上又是什么情况呢？事实上在圆环上是把圆环看成两段，分别移动。那么如何分段，答案是尽量使两段长度相等。证明如下

设总长度为n，分段后的长度分别为a和b，那么得到

$$\begin{aligned} f(a,b) &= \frac{a(a-1)}{2} + \frac{b(b-1)}{2} \\ &= \frac{a(a-1)}{2} + \frac{(n-a)(n-a-1)}{2} \\ &= a^2 - na + \frac{n^2-n}{2} \end{aligned}$$

要使$f(a,b)$取得最大值，那么取

$$a = \frac{n}{2}$$

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
typedef long long LL;

int main()
{
    int n;
    while(cin>>n)
    {
        int a = n >> 1;
        int b = n - a;
        int ans = a * (a - 1) / 2 + b * (b - 1) / 2;
        cout<<ans<<endl;
    }
    return 0;
}
```

题目： http://codeforces.com/problemset/problem/248/B

题意： 找出一个最小的n位数，使得它的素因子分解中仅含有2，3，5，7，如果不存在则输出-1。

分析： 首先n小于4的情况很好判断，接下来只研究n大于等于4的情况。我们可以发现$105 \times 10^{n-3}$满足素因子分解中仅含有2，3，5，7的n位数，但不是最小的，当然这个可以继续减去210，直到最小即可。即

$$x = 105 \times 10^{n-3} - 210k$$

变换一下，可以得到最小的n位数为

$$x = 10^{n-1} + 5 \times 10^{n-3} mod\ 210$$

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;

int main()
{
    int n;
    while(cin>>n)
    {
        if(n < 3)
        {
            puts("-1");
            continue;
        }
        if(n == 3)
        {
            puts("210");
            continue;
        }
        printf("1");
        int ans = 50;
        for(int i = 1; i <= n - 4; i++)
        {
            printf("0");
            ans = ans * 10 % 210;
        }
        printf("%03d\n", ans);
    }
    return 0;
}
```

题目： UVA10025，给定$?1?2?3...?n = k$，其中?代表正负号，求满足条件最小的n。

分析： 在上述序列中，设所有正数之和为a，所有负数之和为b，那么得到

$$a - b = \frac{n(n+1)}{2}$$
$$a + b = k$$

继续消去a得到

$$-2b = \frac{n(n+1)}{2} - k \geq 0$$

接下来往上枚举即可。

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;

int main()
{
    int T;
    scanf("%d", &T);
    for(int i = 1; i <= T; i++)
    {
        int k;
        scanf("%d", &k);
        if(i > 1) puts("");
        if(k < 0) k = -k;
        int ans = 0;
        for(int n = 1; ;n++)
        {
            ans += n;
            if(ans >= k && (ans - k) % 2 == 0)
            {
                printf("%d\n", n);
                break;
            }
        }
    }
    return 0;
}
```

题目： http://acm.hdu.edu.cn/showproblem.php?pid=2563

题意： 在一个无限大的二维方格平面中，每次只能移动一个格子，每次只能向上或左右移动，并且踩过的格子会消失。
求走n步一共有多少种方案。

分析： 设a_n为第n步向上的方案数，b_n为第n步向左或向右的方案数，f_n为当前的总方案数。

因为向上只有一个方向，所以$a_n = a_{n-1} + b_{n-1}$，而之前向上的可以走两个方向，之前向左或者右的只能按原来的方向继续走，所以$b_n = 2a_{n-1} + b_{n-1}$，总数为$f_n = a_n + b_n$，最终得到$f_n = 2f_{n-1} + f_{n-2}$。

代码：

[cpp] view plain copy

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
typedef long long LL;
const int N = 25;

LL dp[N];

void Init()
{
    dp[1] = 3;
    dp[2] = 7;
    for(int i = 3; i < N; i++)
        dp[i] = 2 * dp[i - 1] + dp[i - 2];
}

int main()
{
    int T;
    Init();
    scanf("%d", &T);
    while(T--)
    {
        int n;
        scanf("%d", &n);
        printf("%lld\n", dp[n]);
    }
    return 0;
}
```

题目： http://www.lydsy.com/JudgeOnline/problem.php?id=2659

题意： 给定两个奇质数p和q，求如下表达式的值。

$$g(p,q) = \sum_{k=1}^{\frac{p-1}{2}}\lfloor\frac{kq}{p}\rfloor + \sum_{k=1}^{\frac{q-1}{2}}\lfloor\frac{kp}{q}\rfloor$$

分析： 考虑如下直线

$$y = \frac{q}{p}x$$

那么，题目中表达式的两部分分别代表本直线下方和左方整数点的数目，如果p与q互质，则这条直线上没有整点，所以不会重复，当然p与q相等时也是一种情况，最终答案为矩形整点数目的四分之一。

```cpp
#include <iostream>
#include <string.h>
#include <stdio.h>

using namespace std;
typedef long long LL;

int main()
{
    LL p, q;
    while(cin>>p>>q)
    {
        if(p == q) cout<<p * q / 4<<endl;
        else cout<<(p - 1) * (q - 1) / 4<<endl;
    }
    return 0;
}.
```

递推关系属于2阶常系数线性齐次递推关系,因此可以利用特征方程求解通项