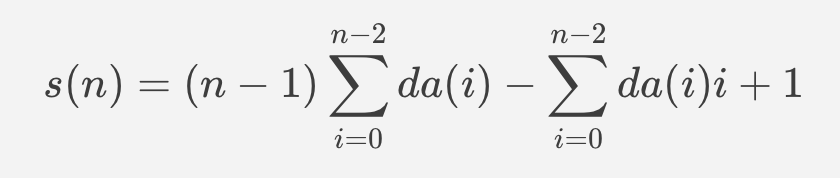
**傻逼数学结论&&定理**

**Lindström–Gessel–Viennot lemma**

路径不相交定理

dag中n个起点 n个终点 矩阵a[i][j]表示第i个起点到第j个终点路径方案数，整个矩阵的行列式就是这n条线路不相交的方案数

**阿贝尔变换**

****

求二次查分的求和

1. sum[i] = a[1]+...+a[i] + delta[1]\*i + delta[2]\*(i - 1) + delta[3]\*(i - 2)+...+delta[i]\*1 *// a[i]为原始数组*
2. = sigma( a[x] ) + sigma( delta[x] \* (i + 1 - x) )
3. = sigma( a[x] ) + (i + 1) \* sigma( delta[x] ) - sigma( delta[x] \* x )

**错排问题**

长度为n的排列满足对任意的i p[i]!=i 排列的个数

两个公式：

**D(n)=(n-1) \*(D(n-1) + D(n-2))**

**D(n)=n!\*(1/2 -1/3! +1/4!- 1/5!+ ··· ··· +((-1)^(n-1))/(n-1)!+((-1)^n)/n! )**

**威尔逊定理及其逆定理**

若p为质数，则p|(p-1)!+1 这个结论的逆定理也成立

亦：(p-1)! ≡ p-1 ≡ -1(mod p)

对于1~p-1 的逆是一个1~p-1排列 除了 1 和 p-1他们的逆为自己本身，其余的可以两两配对 x\*y % p=1，定理得证。

有(p-1)! p在式子中出现时可以考虑威尔逊定理。

**费马小定理**

假如p是质数，且gcd(a,p)=1，那么

a^(p-1) ≡1（mod p）

我们可以利用费马小定理来简化幂模运算：由于a^(p-1)≡a^0≡1(mod p)，所以a^x(mod p)有循环节，长度为p-1，所以a^x≡a^(x%(p-1))(mod p)

**欧拉定理**

若a,m为正整数，**且gcd(a,m) = 1，**则

a^φ(m)≡1(mod m)

**欧拉函数的性质：**

(1)   p^k型欧拉函数:

若N是质数p(即N=p), φ(n)= φ(p)=p-p^(k-1)=p-1。

若N是质数p的k次幂(即N=p^k)，φ(n)=p^k-p^(k-1)=(p-1)p^(k-1)。

(2)mn型欧拉函数

设n为正整数，以φ(n)表示不超过n且与n互素的正整数的个数，称为n的欧拉函数值。若m,n互质，φ(mn)=(m-1)(n-1)=φ(m)φ(n)。

(3)特殊性质:

若n为奇数时，φ(2n)=φ(n)。

对于任何两个互质 的正整数a,n(n>2)有:a^φ(n)=1 mod n (恒等于)此公式即 **欧拉定理**

当n=p 且 a与素数p互质(即:gcd(a,p)=1)则上式有: a^(p-1)=1 mod n (恒等于)此公式即 **费马小定理**

**求幂大法（广义欧拉定理）及其证明**

对于同余式a^b≡x(mod m)，如何求出x？（1<=a,m<=10^9，1<=b<=10^1000000）

注意到b很大，我们可以先采取一些方法降幂。

若gcd(a, m)=1，那么使用欧拉定理即可：a^b≡a^(b%φ(m))(mod m)

若gcd(a,m)>1，且b>φ(m)，则有“求幂大法”——a^b≡a^(b%φ(m)+φ(m))(mod m)

（当b<=φ(m)时直接用快速幂即可）

**对集合的dp**

for(int x = 1;x < (1<<n);x++)

for(int i = x&(x-1);i > 0;i = x&(i-1))

x是从小到大枚举n的子集，i是枚举x的真子集。

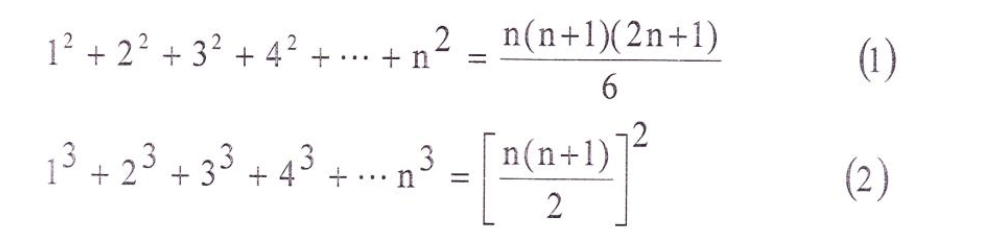
**圆上撒n个点，求最多块数**

1, 2, 4, 8, 16, 31, 57, 99, 163, 256, 386…

a(n) = C(n-1, 4) + C(n-1, 3) + ... + C(n-1, 0) = C(n, 4) + C(n-1, 2) + n.

a(n) = 5\*a(n-1) - 10\*a(n-2) + 10\*a(n-3) - 5\*a(n-4) + a(n-5), n > 4.

**平方&立方和公式**



**树状数组求第k大**

int findkth(int k)

{

int ans=0,cnt=0;

for (int i=20;i>=0;--i)

{

ans+=1<<i;

if (ans>=n||cnt+c[ans]>=k)ans-=1<<i;

else cnt+=c[ans];

}

return ans+1;

}

**博弈论**

**(一)巴什博奕(Bash Game):**

只有一堆n个物品,两个人轮流从这堆物品中取物,规定每次至少取一个,最多取m个.最后取光者得胜.

若(m+1) | n，则先手必败，否则先手必胜。

显然,如果n=m+1,那么由于一次最多只能取m个,所以,无论先取者拿走多少个,后取者都能够一次拿走剩余的物品,后者取胜.因此我们发

现了如何取胜的法则：如果n=(m+1)r+s,(r为任意自然数,s≤m),那么先取者要拿走s个物品,如果后取者拿走k(≤m)个,那么先取者再拿

走m+1-k个,结果剩下(m+1)(r-1)个,以后保持这样的取法,那么先取者肯定获胜.总之,要保持给对手留下(m+1)的倍数,就能最后获胜.

**(二)威佐夫博奕(Wythoff Game):**

有两堆各若干个物品,两个人轮流从某一堆或同时从两堆中取同样多的物品,规定每次至少取一个,多者不限,最后取光者得胜.

奇异局势下先手必败，非奇异局势下先手必胜。

这种情况下是颇为复杂的.我们用(ak,bk)(ak ≤bk ,k=0,1,2,...,n)表示两堆物品的数量并称其为局势,如果甲面对(0,0),那么甲已经

输了,这种局势我们称为奇异局势.前几个奇异局势是：(0,0)、(1,2)、(3,5)、(4,7)、(6,10)、(8,13)、(9,15)、(11,18)、(12,20).

可以看出,a0=b0=0,ak是未在前面出现过的最小自然数,而bk= ak + k,奇异局势有如下三条性质：

1、任何自然数都包含在一个且仅有一个奇异局势中.

由于ak是未在前面出现过的最小自然数,所以有ak > ak-1 ,而bk= ak + k > ak-1 + k-1 = bk-1 > ak-1 .所以性质1.成立.

2、任意操作都可将奇异局势变为非奇异局势.

事实上,若只改变奇异局势(ak,bk)的某一个分量,那么另一个分量不可能在其他奇异局势中,所以必然是非奇异局势.如果使(ak,bk)的两

个分量同时减少,则由于其差不变,且不可能是其他奇异局势的差,因此也是非奇异局势.

3、采用适当的方法,可以将非奇异局势变为奇异局势.

假设面对的局势是(a,b),若b = a,则同时从两堆中取走a 个物体,就变为了奇异局势(0,0)；如果a = ak ,b > bk,那么,取走b - bk个物

体,即变为奇异局势；如果a = ak , b < bk ,则同时从两堆中拿走ak - ab - ak个物体,变为奇异局势( ab - ak , ab - ak+ b - ak)

；如果a > ak ,b= ak + k,则从第一堆中拿走多余的数量a - ak 即可；如果a < ak ,b= ak + k,分两种情况,第一种,a=aj (j < k),从

第二堆里面拿走b - bj 即可；第二种,a=bj (j < k),从第二堆里面拿走b - aj 即可.

从如上性质可知,两个人如果都采用正确操作,那么面对非奇异局势,先拿者必胜；反之,则后拿者取胜.

那么任给一个局势(a,b),怎样判断它是不是奇异局势呢？我们有如下公式：

ak =[k(1+√5)/2](下取整), bk= ak + k (k∈N)

奇妙的是其中出现了有关黄金分割数的式子：(1+√5)/2 =1.618...,若两堆物品个数分别为x,y(x<y)，则k=y-x，再判断x是否等于[(y-x)\*( √5+1)/2] 即可得知是否是奇异局势。

参考例题：POJ1067取石子游戏

参考代码：

var a,b:longint;begin repeat readln(a,b); if a>b then begin a:=a xor b; b:=a xor b; a:=a xor b; end; if a=trunc((b-a)\*(sqrt(5)+1)/2) then writeln(0) else writeln(1); until seekeof;end.

**(三)尼姆博奕(Nimm Game):**

有三堆各若干个物品,两个人轮流从某一堆取任意多的物品,规定每次至少取一个,多者不限,最后取光者得胜.

这种情况最有意思,它与二进制有密切关系,我们用(a,b,c)表示某种局势,首先(0,0,0)显然是奇异局势,无论谁面对奇异局势,都必然失败.第二种奇异局势是(0,n,n),只要与对手拿走一样多的物品,最后都将导致(0,0,0).仔细分析一下,(1,2,3)也是奇异局势,无论对手如何拿,接下来都可以变为(0,n,n)的情形.

计算机算法里面有一种叫做按位模2加,也叫做异或的运算,我们用符号xor表示这种运算.这种运算和一般加法不同的一点是1+1=0.先看

(1,2,3)的按位模2加的结果：

1 =二进制01

Xor 2 =二进制10

Xor 3 =二进制11

———————

0 =二进制00

对于奇异局势(0,n,n)也一样,结果也是0.

任何奇异局势(a,b,c)都有a xor b xor c =0。该结论可以推广至若干堆，都是成立的。

如果我们面对的是一个非奇异局势(a,b,c),要如何变为奇异局势呢？假设a < b< c,我们只要将c 变为a xor b,即可,因为有如下的运算

结果: a xor b xor (a xor b)=(a xor a) xor (b xor b)=0 xor 0=0.要将c 变为a xor b,只要从c中减去c-(a xor b)即可.

**(四)Nim Staircase博奕:**

这个问题是尼姆博弈的拓展：游戏开始时有许多硬币任意分布在楼梯上，共n阶楼梯从地面由下向上编号为0到n。游戏者在每次操作时

可以将楼梯j(1<=j<=n)上的任意多但至少一个硬币移动到楼梯j-1上。游戏者轮流操作，将最后一枚硬币移至地上（0号）的人获胜。

算法：将奇数楼层的状态异或，和为0则先手必败，否则先手必胜。证明略。

例题：Poj1704

这道题可以把两个棋子中间间隔的空格子个数作为一堆石子，则原题转化为每次可以把左边的一堆石子移到相邻的右边的一堆中。也就

是阶梯尼姆博弈，注意对输入数据先排序，然后倒着往前数（a[n]-a[n-1]-1为第一个），奇数个数到的就做一下xor，其中最前面的看

做a[1]-0-1，参考程序：

var t,n,b,i,j:longint; a:array[0..1000]of longint;begin readln(t); repeat dec(t); readln(n); for i:=1 to n do read(a[i]); qsort(1,n);//快排略 j:=0; b:=0; for i:=n downto 1 do begin inc(j); if odd(j) then b:=b xor (a[i]-a[i-1]-1); end; if b=0 then writeln('Bob will win') else writeln('Georgia will win'); until t=0;end.

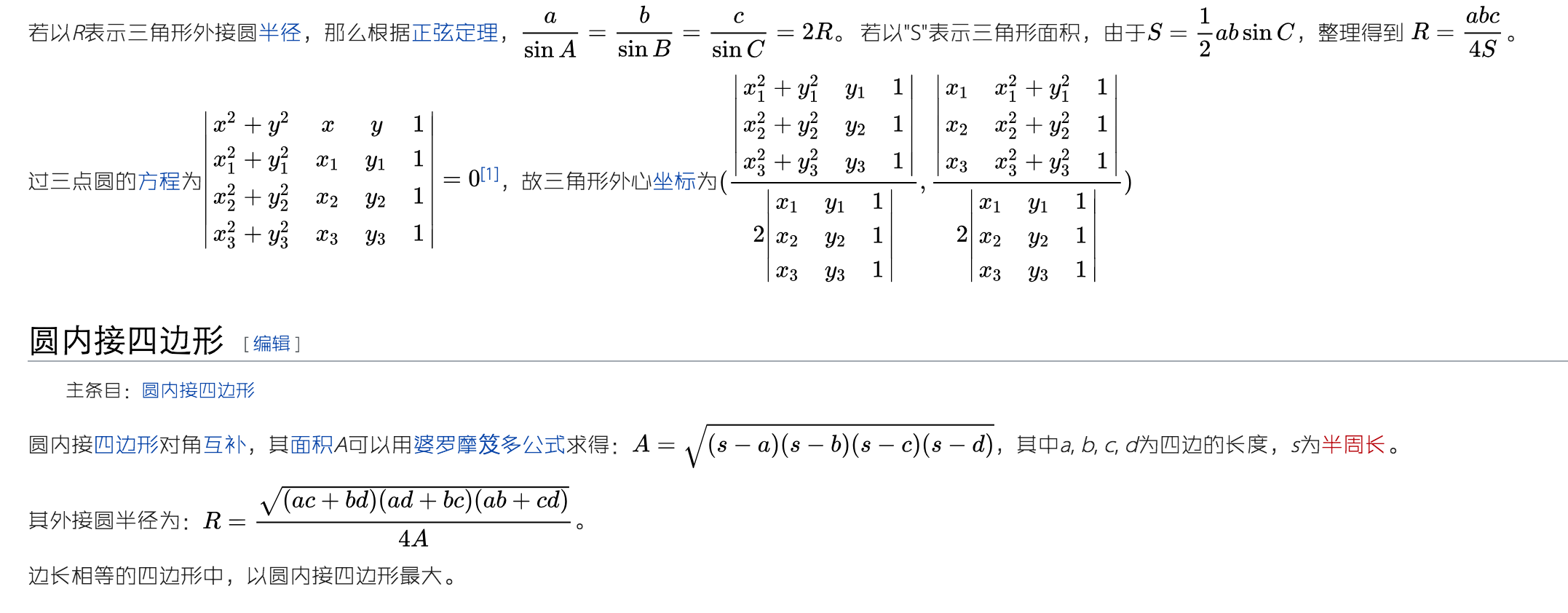
**（五）斐波那契博弈：**

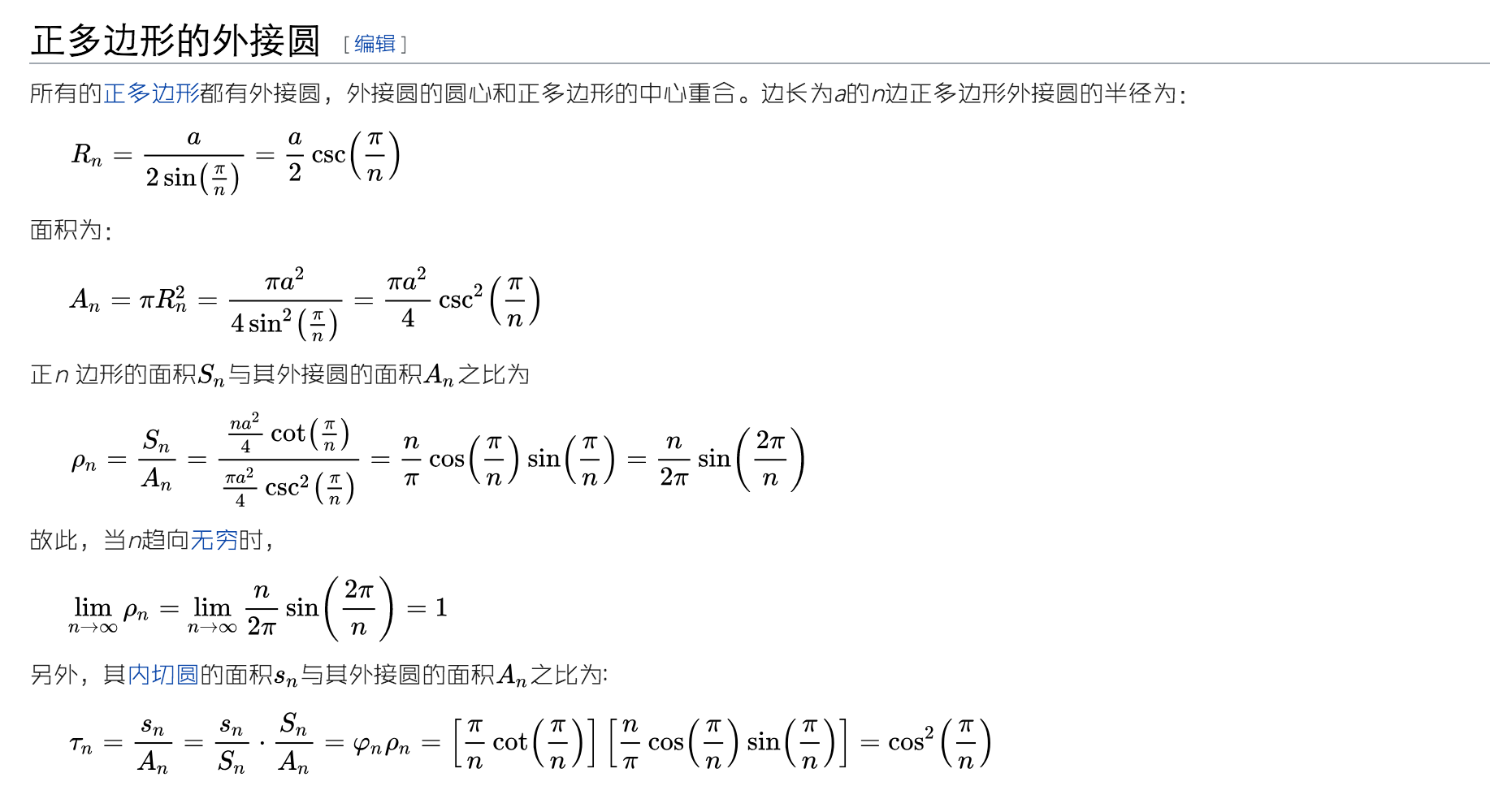
有一堆物品，两人轮流取物品，先手最少取一个，至多无上限，但不能把物品取完，之后每次取的物品数不能超过上一次取的物品数的二倍且至少为一件，取走最后一件物品的人获胜。

结论是：先手胜当且仅当n不是斐波那契数（n为物品总数）

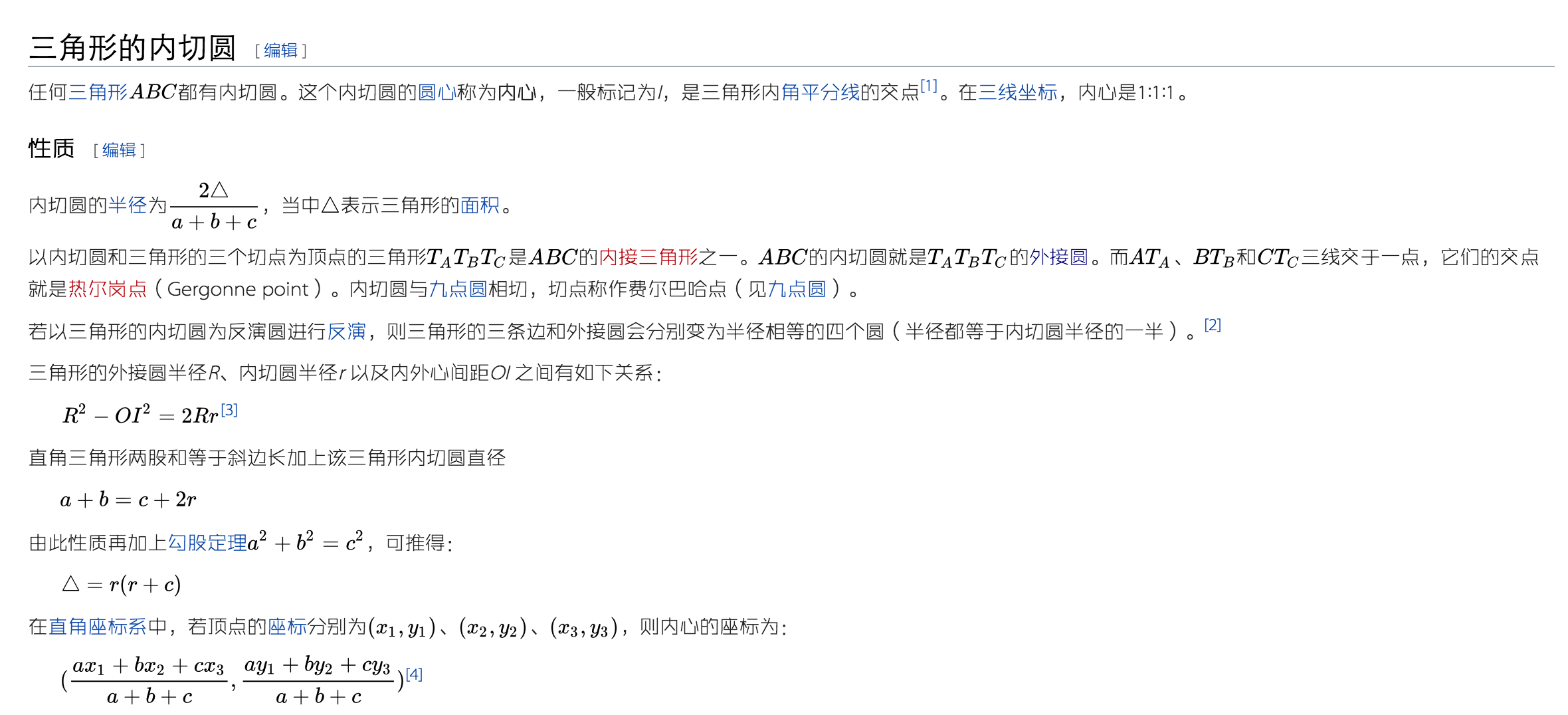
**判断C(n, k)的奇偶性，当且仅当n & k == k 时**

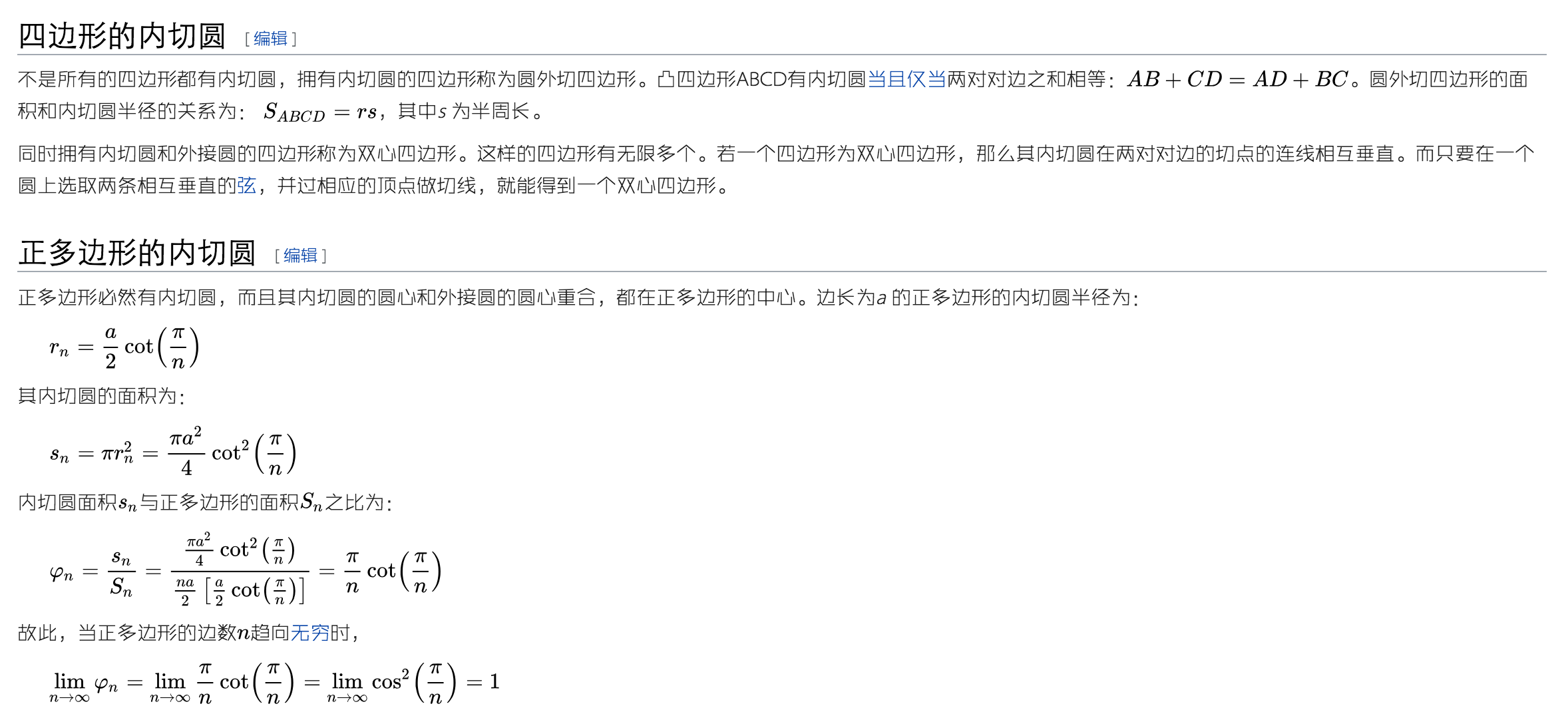
**外接圆**

****

****

**内切圆**

****

****