各种分割问题
（hdu 2050 &1290）

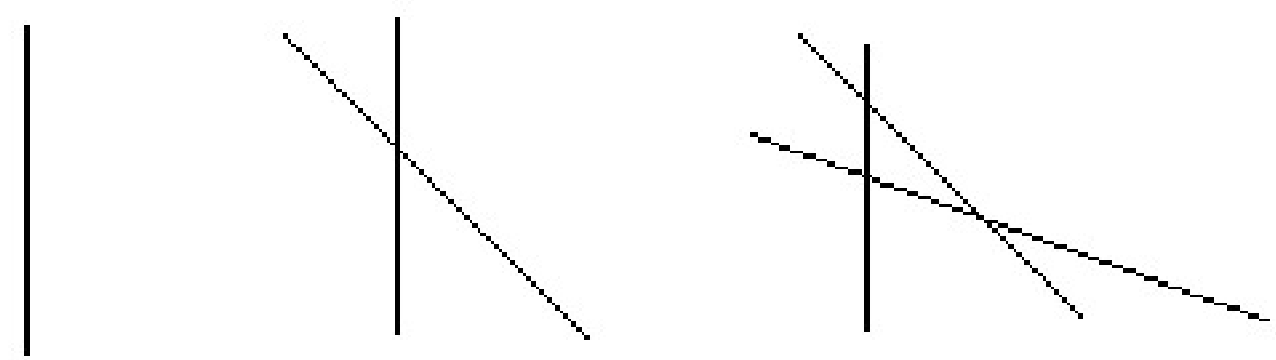
计10-5　　常玉虎

2011年7月27日

(1) n条直线最多分平面问题

- 简述：n条直线最多可以把平面分为多少个区域

- 分析：当有n-1条直线时，平面最多被分成了f（n-1）个区域。则第n条直线要是切成的区域数最多，就必须与每条直线相交且不能有同一交点。 这样就会得到n-1个交点。这些交点将第n条直线分为2条射线和n-2条线断。而每条射线和线断将以有的区域一分为二。这样就多出了2+（n-2）个区域。



- 递推：$f(n)=f(n-1)+n$
- 边界：$f(1)=2$

(1) n条直线最多分平面问题

```
int f(int n)
{   if (1==n) return 2;
    return f(n-1) + n;
}

int main()
{   int n;
    cin >> n;
    cout << f(n) << endl;
    return 0;
}
```

(1) n条直线最多分平面问题

- $f(n)=f(n-1)+n$
- $\quad =f(n-2)+(n-1)+n$
- $\quad \ldots\ldots$
- $\quad =f(1)+1+2+\ldots\ldots+n$
- $\quad =n(n+1)/2+1$

(1) n条直线最多分平面问题

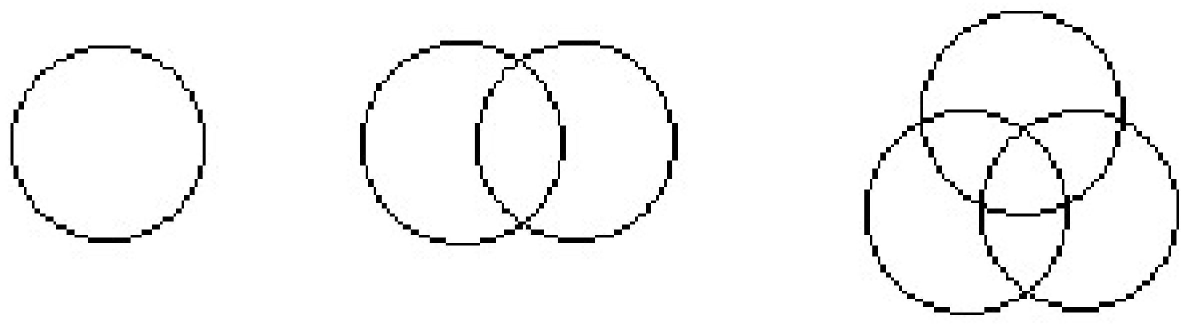
```
#include <iostream>
using namespace std;

int main()
{
    int n;
    cin >> n;
    cout << n*(n+1)/2+1 << endl;
    return 0;
}
```

(2) 封闭曲线分平面问题

- 简述：如设有n条封闭曲线画在平面上，而任何两条封闭曲线恰好相交于两点，且任何三条封闭曲线不相交于同一点，问这些封闭曲线把平面分割成的区域个数。

- 分析：当n-1个圆时，区域数为f(n-1).那么第n个圆就必须与前n-1个圆相交，则第n个圆被分为2（n-1）段线段，增加了2（n-1）个区域。

- 递推：$f(n)=f(n-1)+2(n-1)$
- 边界：$f(1)=2$



(2) 封闭曲线分平面问题

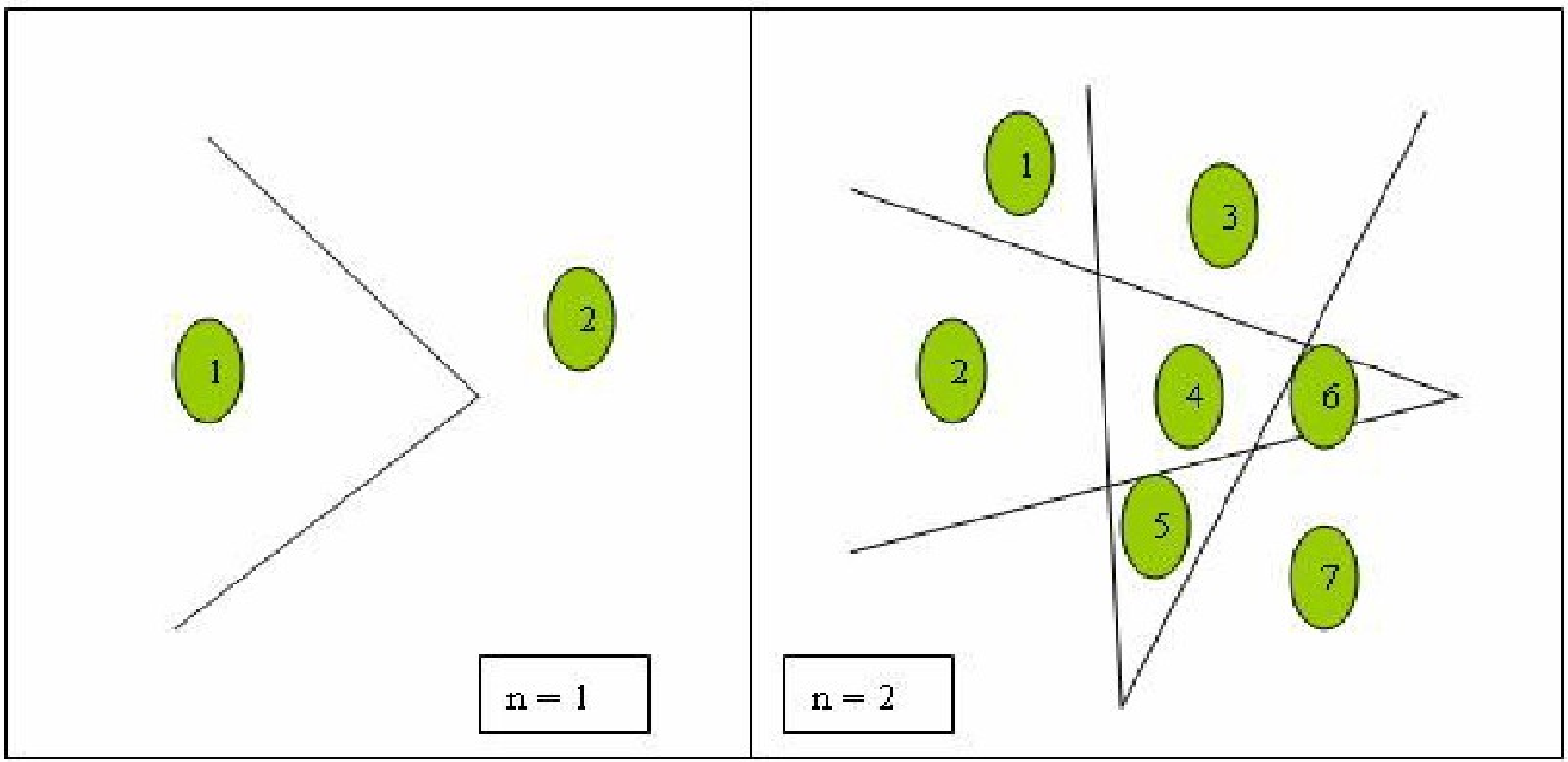
```
int f(int n)
{   if (1==n) return 2;
    return f(n-1) + 2 * (n-1);
}

int main()
{   int n;
    cin >> n;
    cout << f(n) << endl;
    return 0;
}
```

(2) 封闭曲线分平面问题

- $f(n)=f(n-1)+2(n-1)$
- $\quad =f(1)+2+4+\ldots\ldots+2(n-1)$
- $\quad =n^2-n+2$

```
int main()
{
    int n;
    cin >> n;
    cout << n*n-n+2 << endl;
    return 0;
}
```

(3)折线分割平面（hdu2050）

- 问题简述：n条折线分割平面的最大数目。比如，一条折线可以将平面分成两部分，两条折线最多可以将平面分成7部分，具体如下所示。
- 

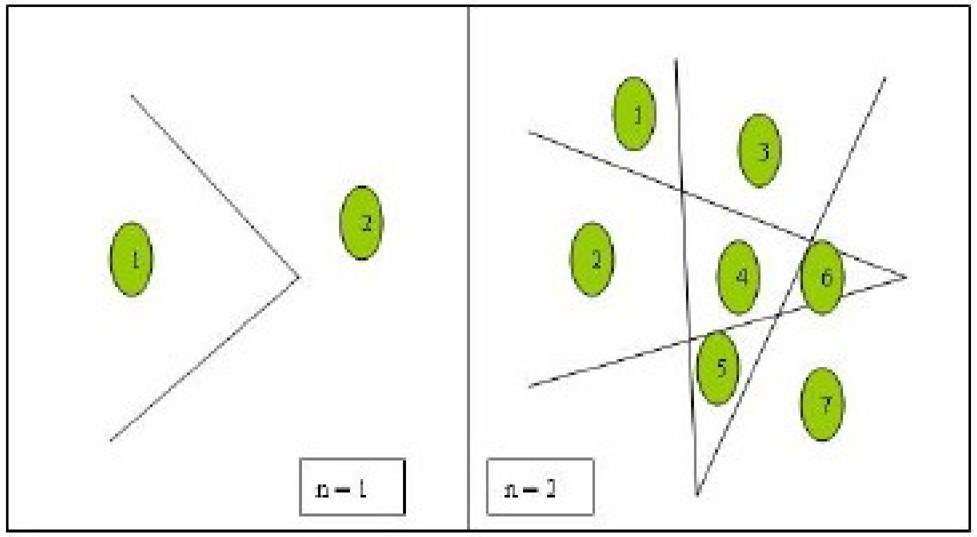
(3)折线分割平面（hdu2050）

- Input：
- 输入数据的第一行是一个整数C,表示测试实例的个数，然后是C 行数据，每行包含一个整数n(0<n<=10000),表示折线的数量。

- Output：
- 对于每个测试实例，请输出平面的最大分割数每个实例的输出占一行。

(3)折线分割平面（hdu2050）

- 分析：根据直线分平面可知，由交点决定了射线和线段的条数，进而决定了新增的区域数。当n-1条折线时，区域数为f（n-1）。为了使增加的区域最多，则折线的两边的线段要和n-1条折线的边，即2*（n-1）条线段相交。那么新增的线段数为4*（n-1），射线数为2。但要注意的是，折线本身相邻的两线段只能增加一个区域。



- 递推：$f(n)=f(n-1)+4(n-1)+2-1$
- 边界：$f(1)=2$

(3)折线分割平面（hdu2050）

```
int f(int n)
{   if (1==n) return 2;
    return f(n-1) + 4 * (n-1) + 1;
}

int main()
{   int n;
    cin >> n;
    cout << f(n) << endl;
    return 0;
}
```

(3)折线分割平面（hdu2050）

- $f(n)=f(n-1)+4(n-1)+2-1$
- $=f(n-1)+4(n-1)+1$
- $=f(n-2)+4(n-2)+4(n-1)+2$
- ……
- $=f(1)+4+4*2+\ldots\ldots+4(n-1)+(n-1)$
- $=2n^2-n+1$

(3)折线分割平面（hdu2050）

```
//Accepted 2050 0MS 300K 161B
#include <iostream>
using namespace std;
int main()
{   int n, t;
    cin >> t;
    while (t--)
    {       cin >> n;
            cout << 2*n*n-n+1 << endl;}
    return 0;
}
```

(4)平面分割空间（hdu1290）

- 由二维的分割问题可知，平面分割与线之间的交点有关，即交点决定射线和线段的条数，从而决定新增的区域数。试想在三维中则是否与平面的交线有关呢？当有n-1个平面时，分割的空间数为f（n-1）。要有最多的空间数，则第n个平面需与前n-1个平面相交，且不能有共同的交线。即最多有n-1 条交线。而这n-1条交线把第n个平面最多分割成g（n-1）个区域。（g（n）为（1）中的直线分平面的个数 ）此平面将原有的空间一分为二，则最多增加g（n-1）个空间。

(4)平面分割空间（hdu1290）

- 递推：$f=f(n-1)+g(n-1)$
- $g(n)=n(n+1)/2+1$
- 边界：$f(1)=2$

(4)平面分割空间（hdu1290）

```
int f(int n)
{   if (1==n) return 2;
    return f(n-1) + n*(n-1)/2+1;
}

int main()
{   int n;
    cin >> n;
    cout << f(n) << endl;
    return 0;
}
```

(4)平面分割空间（hdu1290）

- $f=f(n-1)+g(n-1)$　　ps: $g(n)=n(n+1)/2+1$
- $=f(n-2)+g(n-2)+g(n-1)$
- ……
- $=f(1)+g(1)+g(2)+\ldots\ldots+g(n-1)$
- $=2+(1*2+2*3+3*4+\ldots\ldots+(n-1)n)/2+（n-1）$
- $=(1+2^2+3^2+4^2+\ldots\ldots+n^2-1-2-3-\ldots\ldots-n)/2+n+1$
- $=(n^3+5n)/6+1$

(4)平面分割空间（hdu1290）

- f(n)的另一种直接求法：

- f(n) =g(n) + C_n^3
- = n(n+1)/2+1+n(n-1)(n-2)/6
- =(n^3+5n)/6+1

(4)平面分割空间（hdu1290）

```
//Accepted 1290 0MS 300K 145B
#include <iostream>
using namespace std;

int main()
{
    int n;
    while (cin >> n)
            cout << (n*n*n + 5*n)/6 + 1 << endl;
    return 0;
}
```