字母组合解题报告

方法一：回溯法

回溯是最简单的解题方法。

……

|  |
| --- |
|  |
|  |
|  |
|  |
| C |
| A |
|  |
|  |
|  |
| C |
| B |
| A |
|  |
|  |
|  |
|  |
| B |
| A |
|  |
|  |
|  |
|  |
|  |
| A |

以n=3为例，首先把Ａ进栈，作为第1个组合。然后把B、C进栈，分别作为第2、3个组合，这时无法继续进栈了，所以弹出C，并把B+1变为C。以此类推，直到找到要求的第k个组合。

注意：对于使用系统栈者会爆栈……

程序如下：Charcom1.cpp

方法二：递推法

首先算出分别以字母A、B……开头的组合各有多少个，然后先看要求的组合是哪个字母开头的，然后不看首字母，接续看第二个字母（第二个字母>第一个字母），不断接近要求的组合，最后找到。

例如：n=5,k=7时：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 首字母 | A | B | C | D | E |
| 个数 | 16 | 8 | 4 | 2 | 1 |

1. 因为7<16，所以以A开头；
2. 因为7<8，所以第二个字母为B；
3. 因为4+2<7，所以第三个字母为D；
4. 因为7-4-2=1，所以结束。

输出ABD。

程序如下：Charcom2.cpp

方法三：分治法

|  |  |
| --- | --- |
| 序号 | 组合 |
| 1 | A |
| 2 | AB |
| 3 | ABC |
| 4 | AC |
| 5 | B |
| 6 | BC |
| 7 | C |

**N=3**

分析样例，我们发现，序列的总数是2^n-1，而序列的前2^（n-1）组合首字母为A，接着，我们来看第2到2^（n-1）个组合，分别是B，BC，C，而它们则可以看作是A，AB，B（ASCII码减一），这就是n=2时的序列，接着我们再看后半部分（第2^（n-1）+1到最后一个组合），它们也是B，BC，C，不断二分，就可以在O（nlog2n）的复杂度内找到解。

整理一下以上思想，得出整个算法如下：

1、初始化，读入数据；

2、把当前看做’A’的ASCII码（65）赋值给t；

3、循环

（1）如果k=1，打印chr（t）；退出循环；

（2）如果k<2^（n-1），那么先打印chr（t），然后把n-1（缩小为规模是n-1的子问题），把t+1（把’B’看做是’A’），继续循环（continue）；

（3）如果k>2^（n-1），那么做k-2^（n-1），n-1（缩小为规模是n-1的子问题），继续循环；

4、回行，关闭文件；

程序如下：Charcom3.cpp