Combination

组合

问题：

求拥有n个元素的集合$A=\{a_0,a_1,a_2,\dots,a_{n-1}\}$中任意取$m$个元素（$m\le n$，$m$和$n$都是自然数）的所有组合。

解法：

本文末尾列了很多关于组合算法的文献。本文介绍一种简单易记的算法。

第 1 轮操作，从长度为n的集合$A=\{a_0,a_1,a_2,\dots,a_{n-1}\}$中取出 1 个作为新的集合，设置数组$s=[1_0,0_1,0_2,\dots,0_{n-1}]$表示对集合$A$的选择，第$i$个数字$s_i=1$表示选择数字$a_i$，$s_i=0$表示不选择数字$a_i$。唯一的 1 在数组 s 中选择任意位置，可以得到$C_1^n=n$个组合：

$$[1_0,0_1,0_2,\dots,0_{n-1}]$$
$$[0_0,1_1,0_2,\dots,0_{n-1}]$$
$$[0_0,0_1,1_2,\dots,0_{n-1}]$$
$$\dots\dots$$
$$[0_0,0_1,0_2,\dots,1_{n-1}]$$

第 2 轮操作，取出 2 个元素，可以看作是在上面n个所有组合的基础上增加一个 1。

对于第 1 个数组$[1_0,0_1,0_2,\dots,0_{n-1}]$增加一个 1 后得到数组$[1_0,1_1,0_2,\dots,0_{n-1}]$。原本的$1_0$保持不变，新增的$1_1$可以选择后面等于 0 的$n-1$个位置，生成$n-1$个组合：

$$[1_0,1_1,0_2,0_3,\dots,0_{n-1}]$$
$$[1_0,0_1,1_2,0_3,\dots,0_{n-1}]$$
$$[1_0,0_1,0_2,1_3,\dots,0_{n-1}]$$
$$\dots\dots$$
$$[1_0,0_1,0_2,0_3,\dots,1_{n-1}]$$

需要注意的是，新增的 1 必须在原数组的所有的 1 的后面。对于第 2 个数组$[0_0,1_1,0_2,\dots,0_{n-1}]$，新增的 1 可以选择后面等于 0 的$n-1$个位置，生成$n-2$个组合：

$$[0_0,1_1,1_2,0_3,\dots,0_{n-1}]$$
$$[0_0,1_1,0_2,1_3,\dots,0_{n-1}]$$
$$\dots\dots$$
$$[0_0,1_1,0_2,0_3,\dots,1_{n-1}]$$

如果不注意，让新增的 1 在原数组的任意的 1 的前面，则会产生重复的组合，仍然以第 2 个数组$[0_0,1_1,0_2,\dots,0_{n-1}]$为例，如果新增的 1 可以选择任意等于 0 的位置，会生成$n-1$个组合：

$$[1_0,1_1,0_2,0_3,\dots,0_{n-1}]$$
$$[0_0,1_1,1_2,0_3,\dots,0_{n-1}]$$
$$[0_0,1_1,0_2,1_3,\dots,0_{n-1}]$$
$$\dots\dots$$
$$[0_0,1_1,0_2,0_3,\dots,1_{n-1}]$$

但其中$[1_0,1_1,0_2,0_3,\dots,0_{n-1}]$与第 1 个数组产生的组合重复了。对第 1 轮中所有的数组重复该操作，即可得到选取 2 个元素的所有组合，共有$C_2^n=\frac{n\times(n-1)}{2}$个。

第 3 轮操作，取出 3 个元素，可以看作是在第 2 轮操作的$(n-1)!$个组合基础上增加一个 1，对于之前的每个组合，保持之前的二个 1 不变，新的 1 可以选择原数组中最后一个 1 之后的任意等于 0 的位置。注意新增的 1 不能比原数组中的任意的 1 更靠前，必须在所有的 1 之后的位置进行选择。

重复上述的操作，直到选取m个元素，即可得到所有的组合，算法结束。然后根据s的全排列生成集合A的所有组合即可。该算法时间复杂度为$C_m^n = \frac{n!}{m!(n-m)!}$。

StackOverflow 上关于组合产生算法的问题：

http://stackoverflow.com/questions/127704/algorithm-to-return-all-combinations-of-k-elements-from-n

二项式系数：

https://en.wikipedia.org/wiki/Binomial_coefficient

Chase's Twiddle - Algorithm 382: Combinations of M out of N Objects:

http://dl.acm.org/citation.cfm?id=362502

http://www.netlib.no/netlib/toms/382

Buckles - Algorithm 515: Generation of a Vector from the Lexicographical Index:

http://dl.acm.org/citation.cfm?id=355739

https://www.researchgate.net/profile/Bill_Buckles/publication/220492658_Algorithm_515_Generation_of_a_Vector_from_the_Lexicographical_Index_G6/links/5716d7ad08ae497c1a5706ec.pdf

Remark on algorithm 515: Generation of a vector from the lexicographical index combinations:

http://dl.acm.org/citation.cfm?id=1236470