ĐẠI HỌC BÁCH KHOA – ĐẠI HỌC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN

Môn học

Toán Ứng Dụng
Applied mathematics

Giảng Viên: Phạm Minh Tuấn

Giới thiệu

- Phạm Minh Tuấn
- E-mail: pmtuan@dut.udn.vn
- Tel: 0913230910
- Khoa Công nghệ thông tin –Trường ĐHBK – ĐHĐN

Contents

Bài 5:
Combinatorics

Contents

Introduction

Introduction

- Combinatorics studies methods for counting combinations of objects. Usually, the goal is to find a way to count the combinations efficiently without generating each combination separately.
- As an example, consider the problem of counting the number of ways to represent an integer n as a sum of positive integers. For example, there are 8 representations for 4:
  - 1 + 1 + 1 + 1, 1+1+2, 1+2+1, 2+1+1, 2+2,3+1,1+3, 4

Introduction

- A combinatorial problem can often be solved using a recursive function. In the problem of counting the number of ways to represent an integer *n* as a sum of positive integers, we can define a function *f*(*n*) that gives the number of representations for *n*. The values of the function can be recursively calculated as follows:

$$f(n) = \begin{cases} 1 & n = 0 \\ f(0) + f(1) + \cdots + f(n-1) & n > 0 \end{cases}$$

Introduction

$$f(n) = \begin{cases} 1 & n = 0 \\ f(0) + f(1) + \cdots + f(n-1) & n > 0 \end{cases}$$

- The base case is f (0) = 1, because the empty sum represents the number 0.
- if $n > 0$, we consider all ways to choose the first number of the sum. If the first number is k, there are $f(n - k)$ representations for the remaining part of the sum. Thus, we calculate the sum of all values of the form $f(n - k)$ where $k < n$.
- The first values for the function are:
  - f(0) = 1, f(1) = 1; f(2) = 2; f(3) = 4 and f(4) = 8;
  - $f(n) = 2^{n-1}$.

Binomial coefficients

(Hệ số trong triển khai nhị thức)

Binomial coefficients

- The **binomial coefficient** $\binom{n}{k}$ equals the number of ways we can choose a subset of k elements from a set of n elements.
- For example, $\binom{5}{3} = 10$, because the set {1, 2, 3, 4, 5} has 10 subsets of 3 elements.
  - {1,2,3},{1,2,4},{1,2,5},{1,3,4},{1,3,5},
  - {1,4,5},{2,3,4},{2,3,5},{2,4,5},{3,4,5}

Formula 1

- Binomial coefficients can be recursively calculated as follows:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

- The idea is to fix an element x in the set. If x is included in the subset, we have to choose $k-1$ elements from $n-1$ elements, and if x is not included in the subset, we have to choose k elements from $n-1$ elements.
- The base cases for the recursion are $\binom{n}{0} = \binom{n}{n} = 1$ because there is always exactly one way to construct an empty subset and a subset that contains all the elements.

Formula 2

- Another way to calculate binomial coefficients is as follows:

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$$

- There are $n!$ permutations of n elements. We go through all permutations and always include the first k elements of the permutation in the subset. Since the order of the elements in the subset and outside the subset does not matter, the result is divided by $k!$ and $(n-k)!$

Properties

- For binomial coefficients,

$$\binom{n}{k} = \binom{n}{n-k}$$

- because we actually divide a set of n elements into two subsets: the first contains k elements and the second contains $n-k$ elements.
- The sum of binomial coefficients is

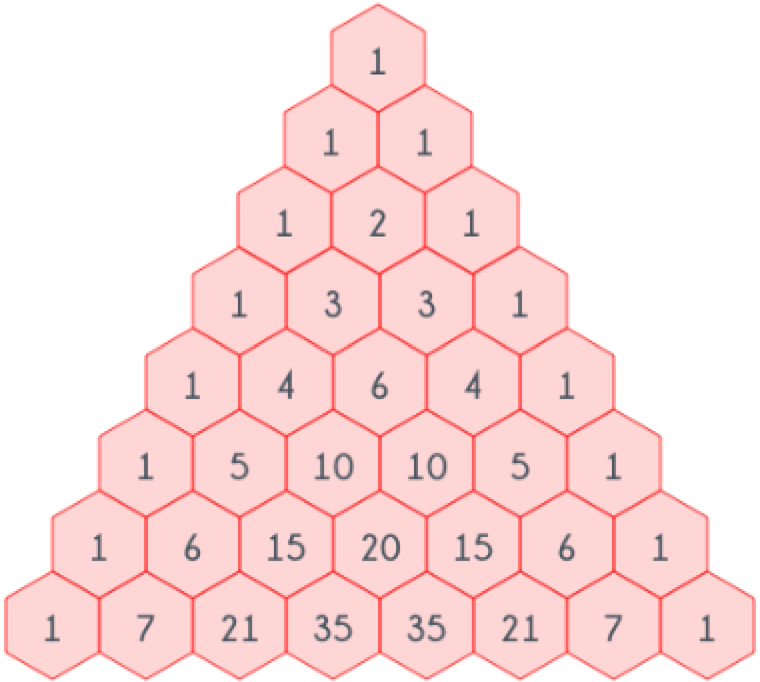
$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n$$

- The reason for the name ”binomial coefficient” can be seen when the binomial $(a-b)$ is raised to the nth power:

$$(a-b)^n = \binom{n}{0} a^n b^0 + \binom{n}{1} a^{n-1} b^1 + \cdots + \binom{n}{n} a^0 b^n$$

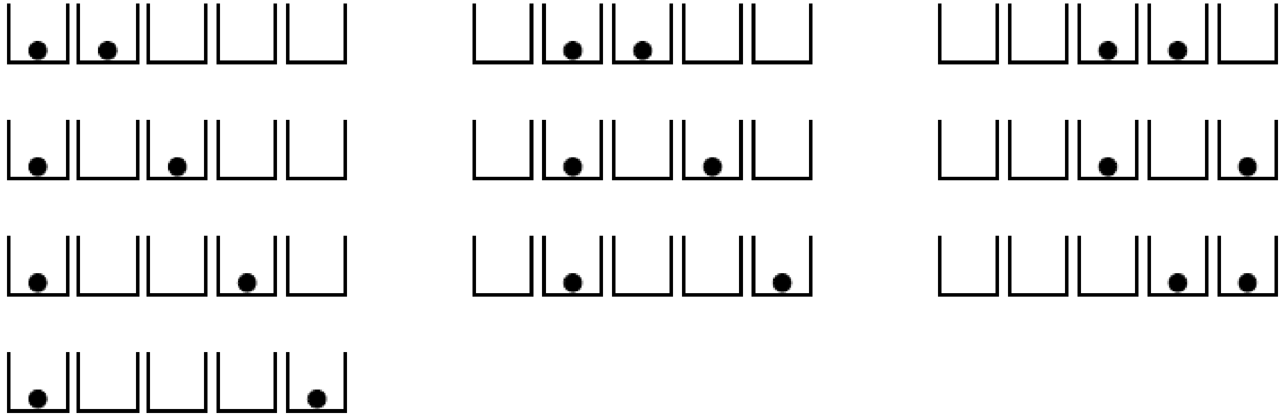
Pascal's triangle

- Binomial coefficients also appear in **Pascal's triangle** where each value equals the sum of two above values:



Boxes and balls

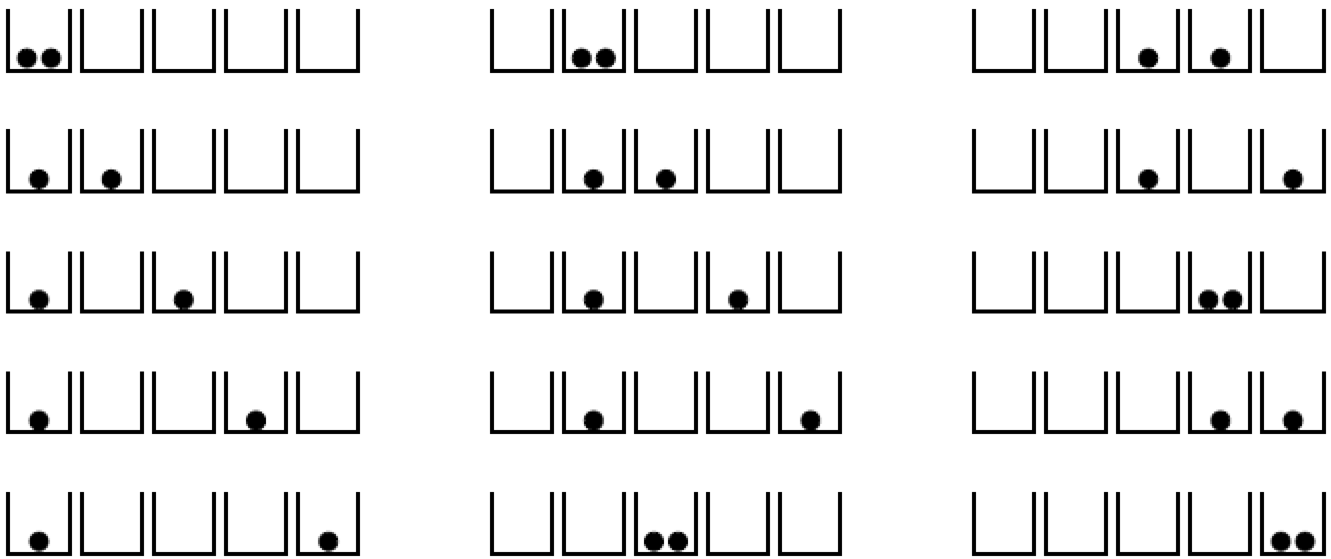
- “**Boxes and balls**” is a useful model, where we count the ways to place k balls in n boxes. Let us consider three scenarios:
- Scenario 1: Each box can contain at most one ball. For example, when $n = 5$ and $k = 2$, there are 10 solutions:



- In this scenario, the answer is directly the binomial coefficient $\binom{n}{k}$

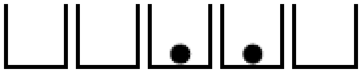
Scenario 2

- Scenario 2: A box can contain multiple balls. For example, when $n = 5$ and $k = 2$, there are 15 solutions:



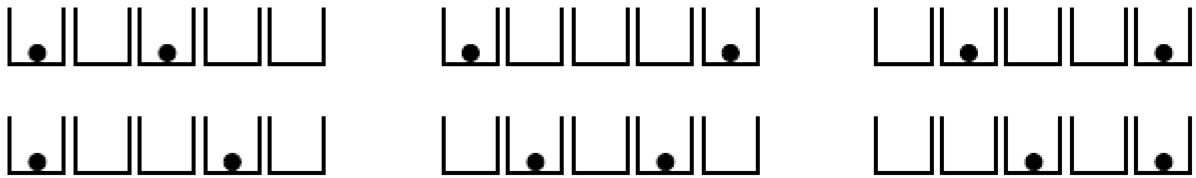
Scenario 2

- The process of placing the balls in the boxes can be represented as a string that consists of symbols ”o” and ”!”. Initially, assume that we are standing at the leftmost box. The symbol ”o” means that we place a ball in the current box, and the symbol ”!” means that we move to the next box to the right.
- Using this notation, each solution is a string that contains k times the symbol ”o” and $n - 1$ times the symbol ”!”. For example, the upper-right solution in the above picture corresponds to the string ”! ! o ! o !”. Thus, the number of solutions is $\binom{n+k-1}{k}$



Scenario 3

- Scenario 3: Each box may contain at most one ball, and in addition, no two adjacent boxes may both contain a ball. For example, when $n = 5$ and $k = 2$, there are 6 solutions:



- In this scenario, we can assume that k balls are initially placed in boxes and there is an empty box between each two adjacent boxes. The remaining task is to choose the positions for the remaining empty boxes. There are $n - 2k + 1$ such boxes and $k + 1$ positions for them. Thus, using the formula of scenario 2, the number of solutions is

$$\binom{n-k+1}{n-2k+1}$$

Multinomial coefficients

- The **multinomial coefficient (hệ số đa thức)**

$$\binom{n}{k_1, k_2, \cdots, k_m} = \frac{n!}{k_1!\,k_2!\cdots k_m!}$$

- equals the number of ways we can divide n elements into subsets of sizes $k_1, k_2, \cdots, k_m$, where $k_1 + k_2 + \cdots + k_m = n$. Multinomial coefficients can be seen as a generalization of binomial cofficients; if m=2, the above formula corresponds to the binomial coefficient formula.

TEST YOUR UNDERSTANDING

- Micro is having an array *A*, having *N* integers. He wants to find out number of ways of choosing exactly *K* even numbers from the array. Can you help Micro with this?
- **Input:**
  - First line consists of two space separated integers denoting N and K.
  - Second line consists of N space separated integers denoting the array A.
- Output:
  - Print number of different ways for Micro to select exactly K even numbers out of the given array.
- Constraints:
  - 1<=N<=10, 1<=A[i]<=100

Solution

```
#include <iostream>
using namespace std;
int f(int n){
    if (n==0) return 1;
    return n*f(n-1);
}
```

```
int main() {
        int n,k,g=0;
        cin >> n >> k;
        for (int i=0;i<n;i++){
            int a;
            cin >> a;
            if (a%2==0) g++;
        }
        if (g<k) cout << 0;
        else {
        int ans = f(g)/(f(k)*f(g-k));
        cout << ans;
        }
}
```

Catalan numbers

Catalan numbers

- The **Catalan number** C_n equals the number of valid parenthesis expressions that consist of n left parentheses and n right parentheses.
- For example, $C_3 = 5$, because we can construct the following parenthesis expressions using three left and right parentheses:
  - ()()()
  - (())()
  - ()(())
  - ((()))
  - (()())

Parenthesis expressions

- The following rules precisely define all valid parenthesis expressions:
  - An empty parenthesis expression is valid.
  - If an expression A is valid, then also the expression (A) is valid.
  - If expressions A and B are valid, then also the expression AB is valid.
- Another way to characterize valid parenthesis expressions is that if we choose any prefix of such an expression, it has to contain at least as many left parentheses as right parentheses. In addition, the complete expression has to contain an equal number of left and right parentheses.

Formula 1

- Catalan numbers can be calculated using the formula

$$C_n = \sum_{i=0}^{n-1} C_i C_{n-i-1}$$

- The sum goes through the ways to divide the expression into two parts such that both parts are valid expressions and the first part is as short as possible but not empty. For any i, the first part contains $i + 1$ pairs of parentheses and the number of expressions is the product of the following values.
  - C_i : the number of ways to construct an expression using the parentheses of the first part, not counting the outermost parentheses

I T F

Formula 2

- Catalan numbers can also be calculated using binomial coefficients

$$C_n = \frac{1}{n+1}\binom{2n}{n}$$

- The formula can be explained as follows:
  - There are a total of $\binom{2n}{n}$ ways to construct a (not necessarily valid) parenthesis expression that contains n left parentheses and n right parentheses. Let us calculate the number of such expressions that are not valid.
  - If a parenthesis expression is not valid, it has to contain a prefix where the number of right parentheses exceeds the number of left parentheses.

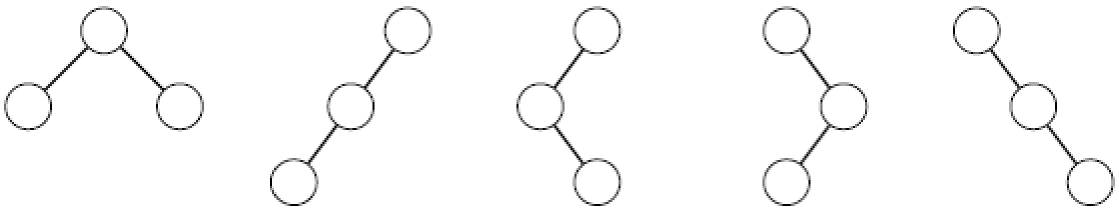
Formula 2

- The idea is to reverse each parenthesis that belongs to such a prefix. For example, the expression ())()(contains a prefix ()), and after reversing the prefix, the expression becomes)((()(.
- The resulting expression consists of $n + 1$ left parentheses and $n - 1$ right parentheses. The number of such expressions is $\binom{2n}{n+1}$, which equals the number of non-valid parenthesis expressions. Thus, the number of valid parenthesis expressions can be calculated using the formula

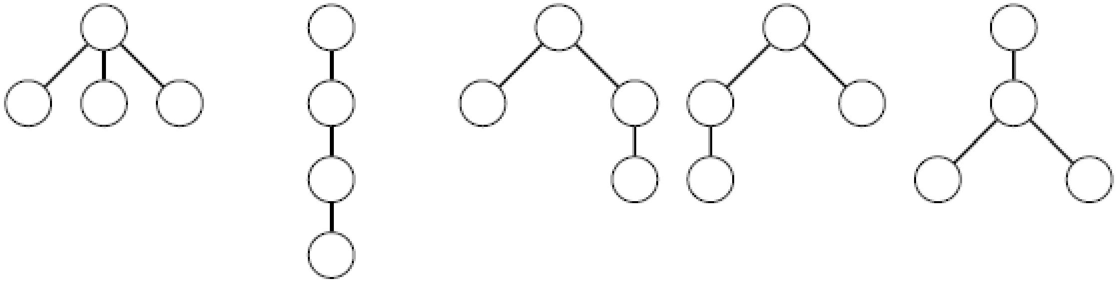
$$\binom{2n}{n} - \binom{2n}{n+1} = \binom{2n}{n} - \frac{n}{n+1}\binom{2n}{n} = \frac{1}{n+1}\binom{2n}{n}$$

Counting trees

- Catalan numbers are also related to trees:
  - there are C_n binary trees of n nodes
  - there are C_{n-1} rooted trees of n nodes
- For example, for $C_3 = 5$, the binary trees are



- and the rooted trees are



TEST YOUR UNDERSTANDING

- Tuan has n bracket pairs and he wants to find out the number of nice expressions he can make out of those bracket pairs. Example : For 3 bracket pairs he can form the following nice expressions ((())), ()(()), ()()(), (())(), (()()) so the answer is 5.
- **Input:**
  - You are given t test cases. Next t lines contains the number n .
- Output:
  - Print the maximum number of NICE expression for each test case in a new line.
- Constraints:
  - $0 < t < 100000$ $0 < n < 100$

Problem