Departamento de Computação

Programação Competitiva unesp

Introdução à Análise Combinatória

Laboratório de Programação Competitiva I

Pedro Henrique Paiola

Rene Pegoraro

Wilson M Yonezawa

Departamento de Computação

Programação Competitiva unesp

Análise Combinatória

- Combinatória, o estudo dos arranjos dos objetos, é uma parte importante da matemática discreta. É o ramo da matemática que se dedica à contagem de elementos ou eventos discretos e de suas possíveis combinações.
- Diversos problemas de programação competitiva envolvem análise combinatória.

Análise Combinatória

- Diversos problemas de contagem e de combinação possuem **soluções fechadas**, ou seja, existem **fórmulas matemáticas** resultantes da análise combinatória que podem ser aplicadas.
- Este é um dos motivos da importância da análise combinatória para a **Computação**, pois permite substituir um algoritmo com complexidade alto (busca por backtracking, por exemplo), por uma única chamada a uma simples fórmula.

Análise Combinatória

- Em Programação Competitiva, isto é particularmente importante, permitindo resolver problemas aparentemente complexos de forma bastante simples, e sem estourar o tempo limite.
- Em alguns casos, também é possível a obtenção de *look-up tables* para soluções *off-line*.

Regras do Produto e da Soma

- Bases da contagem:
  - **Regra do Produto**: Suponha que um procedimento possa ser dividido em uma sequência de duas tarefas. Se houver $\boldsymbol{n}$ formas de fazer a primeira tarefa **e**, para cada uma dessas formas, há $\boldsymbol{m}$ formas de fazer a segunda, então há $\boldsymbol{n.m}$ formas de concluir o procedimento.

Regras do Produto e da Soma

- Bases da contagem:
  - **Regra do Produto**: Suponha que um procedimento possa ser dividido em uma sequência de duas tarefas. Se houver $\boldsymbol{n}$ formas de fazer a primeira tarefa **e**, para cada uma dessas formas, há $\boldsymbol{m}$ formas de fazer a segunda, então há $\boldsymbol{n.m}$ formas de concluir o procedimento.

Exemplo: Quantidade de números de 3 dígitos que podem ser formados apenas com os algarismos $1, 2, 5$ e 7.

Devemos preencher 3 dígitos escolhendo dentro de 4 algarismos: $4 * 4 * 4 = 64$ possibilidades

Se não pudesse haver repetição de algarismos: $4 * 3 * 2 = 24$ possibilidades

Regras do Produto e da Soma

- Bases da contagem:
  - **Regra do Soma**: Se uma tarefa puder ser realizada em uma de $\boldsymbol{n}$ formas **OU** em uma das $\boldsymbol{m}$ formas, em que nenhuma das $\boldsymbol{n}$ formas seja igual a alguma das $\boldsymbol{m}$ formas, então há $\boldsymbol{n} + \boldsymbol{m}$ formas de realizar a tarefa.

Regras do Produto e da Soma

- Bases da contagem:
  - **Regra do Soma**: Se uma tarefa puder ser realizada em uma de $\boldsymbol{n}$ formas **OU** em uma das $\boldsymbol{m}$ formas, em que nenhuma das $\boldsymbol{n}$ formas seja igual a alguma das $\boldsymbol{m}$ formas, então há $\boldsymbol{n} + \boldsymbol{m}$ formas de realizar a tarefa.
  - Caso mais geral: quando há intersecção entre os conjuntos de "formas", devemos subtraí-la da soma:

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

Contagem de associações

- Uma associação é um arranjo de $\boldsymbol{n}$ itens, onde cada item pode ser escolhido de uma lista de $\boldsymbol{m}$ valores, com repetição.
- Por exemplo, quantas formas diferentes existem de se pintar 4 casas utilizando 3 cores.
- Utilizando a regra do produto:

$$S(n, m) = m^n$$

- Existem $S(4{,}3) = 3^4 = 81$ associações possíveis entre 4 casas e 3 cores

Contagem de associações

- Caso específico: **subconjuntos**
  - Quantos subconjuntos podemos formar a partir de um conjunto de $\boldsymbol{n}$ elementos? Trata-se de um problema de seleção sem reposição.
  - A seleção ou não de cada um dos $\boldsymbol{n}$ elementos pode ser representada de forma binária (selecionar ou não selecionar): arranjo binário de $\boldsymbol{n}$ posições.
  - Logo, o número de possíveis subconjuntos é $\boldsymbol{S(n,2) = 2^n}$

Elementos:	x_1	x_2	x_3	x_4	x_5	…	x_n
Selecionar?	1	1	0	0	1	…	0

$2 \cdot 2 \cdot 2 \cdot \cdot \cdot \cdot \cdot \cdot 2 = 2^n$

Contagem de associações

- Exemplo 1:
  - Um computador de 32 bits é capaz de endereçar quantos gigabytes de memória?

$$S(32,2) = 2^{32} = 2^2 . 2^{30} = 4GB$$

- Exemplo 2:
  - Quantas senhas diferentes é possível criar utilizando de 8 a 10 letras ou dígitos, considerando letras minúsculas e maiúsculas.

$$\underbrace{S(8,62)}_{62^8} + S(9,62) + S(10,62) = 853.054.792.520.188.672$$

$62^8 + 62^9 + 62^{10}$

Permutação

- Permutação é um arranjo de $\boldsymbol{n}$ itens, onde cada item aparece exatamente uma única vez.
- O 1º elemento do arranjo pode assumir qualquer um dos $\boldsymbol{n}$ itens, o 2º pode assumir $\boldsymbol{n-1}$ itens (qualquer um, exceto o já assumido pelo 1º) e assim por diante.
- Logo, pela regra do produto:

$$P(n) = n(n-1)(n-2)\ldots 1 = n!$$
$$P(n) = n!$$

Permutação

- Exemplo: anagramas
  - Quantos anagramas existem da palavra MESA
  - Conjunto de elementos: $\{M, E, S, A\}$
  - $P(4) = 4! = 24$ anagramas

Permutação

- Permutação com elementos repetidos
  - Exemplo: anagrama da palavra CASA
  - A letra "A" aparece duas vezes, se aplicássemos a fórmula da permutação, o anagrama **ACSA**, por exemplo, seria contado duas vezes, como se cada letra **A** fosse uma letra diferente. CASA => ACSA, ACSA
  - Considerando um conjunto de elementos, e que o elemento 1 se repete n_1 vezes, o 2 se repete n_2 vezes, e assim por diante, chegamos em:

$$P^{n_1,\ldots,n_n}(n) = \frac{n!}{n_{1!}n_{2!}\ldots n_{n!}}$$

Arranjo

- **Arranjo**: quantas possibilidades há de escolher $\boldsymbol{r}$ elementos de um conjunto de $\boldsymbol{n}$ elementos, em que **a ordem de escolha é relevante**?
- É uma generalização da permutação. Uma permutação pode ser considerada como um arranjo em que $\boldsymbol{r} = \boldsymbol{n}$

$$P(n,r) = \frac{n!}{(n-r)!}$$

Arranjo

```
int PermutationCoeff(int n, int k)
{
    int Fn = 1, Fk;

    for (int i = 1; i <= n; i++)
    {
        Fn *= i;
        if (i == n - k)
            Fk = Fn;
    }
    int coeff = Fn / Fk;
    return coeff;
}
```

Arranjo

$$P(n,r) = \frac{n!}{(n-r)!} = \frac{n(n-1)\ldots(n-r-1)\cancel{(n-r)}!}{\cancel{(n-r)}!}$$

$$P(n,r) = \underbrace{n(n-1)\ldots(n-r-1)}$$

Arranjo

```
int PermutationCoeff(int n, int k)
{
    int coeff = 1;

    for (int i = n; i > (n - k); i--)
        coeff *= i;

    return coeff;
}
```

Departamento de Computação

Programação Competitiva unesp

Combinação

- **Combinação**: quantas possibilidades há de escolher r elementos de um conjunto de n elementos, em que **a ordem de escolha não é relevante**?

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

- Tanto no caso da permutação como da combinação, temos que tomar um pouco de cuidado com nossa implementação (estamos usando muito fatorial), tanto em relação ao tempo quanto ao limite de nossas variáveis.

Combinação

- Para a combinação, pode-se calcular qualquer coeficiente binomial baseado na seguinte recorrência (que deriva o conhecido Triângulo de Pascal):

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

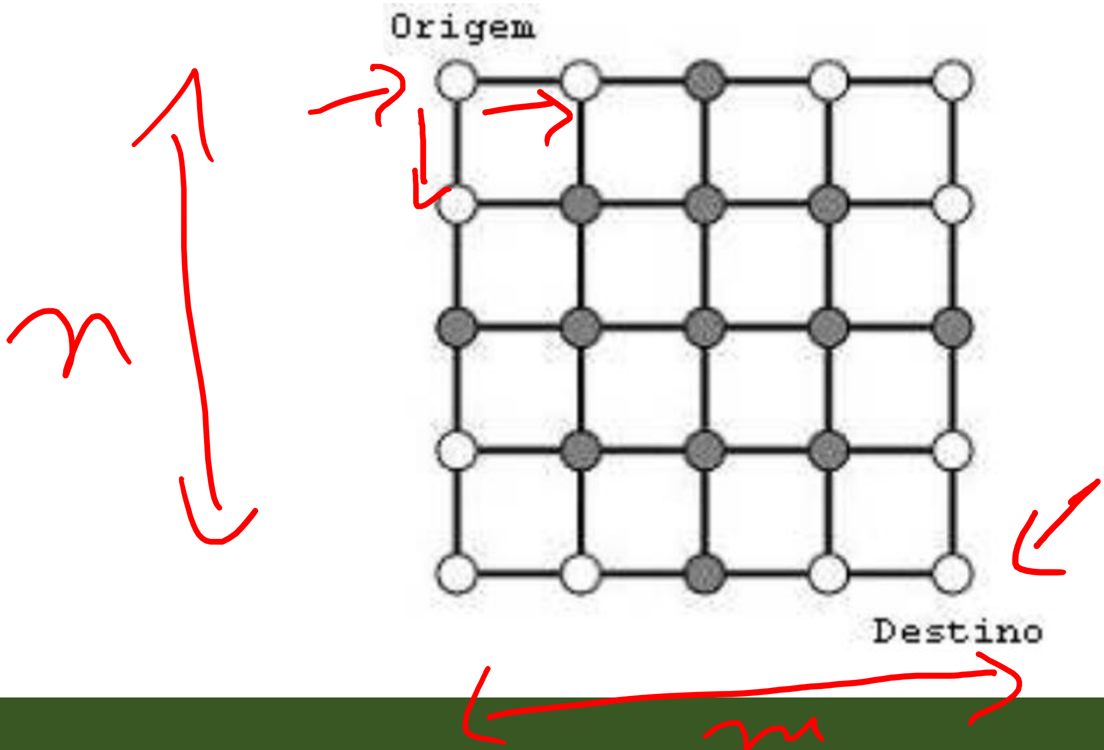
Combinação

```
long long bin[MAXN][MAXR];

void calcularCoefBin(int n, int k){ //Pré-calculando em O(n*k)
    int i, j;
    for (i = 0; i <= n; i++){
        for (j = 0; j <= min(i, k); j++){
            if (j == 0 || j == i)
                bin[i][j] = 1;
            else
                bin[i][j] = bin[i - 1][j - 1] + bin[i - 1][j];
        }
    }
}
```

Exemplo: caminho através de uma grade

- Quantas formas temos de caminhar em uma grade $n \; x \; m$ a partir do canto superior esquerdo e alcançar o canto inferior direito caminhando apenas para baixo e para a direita?



Exemplo: caminho através de uma grade

- **1ª forma de analisar**: note que cada caminho é necessariamente constituído de um conjunto de $n + m$ passos, n para baixo e m para a direita.
  - Sendo assim, um caminho nada mais é do que uma permutação de passos para baixo e para a direita. Podemos considerar como se fosse um anagrama (com “letras” repetidas). Ex: BBDDB. Logo, pela fórmula da permutação com elementos repetidos:

$$P^{n,m}(n,m) = \frac{(n+m)!}{n!\,m!}$$

Exemplo: caminho através de uma grade

- **2ª forma de analisar**: novamente, considerando que um caminho é constituído de $n + m$ passos, n para baixo e m para a direita.
  - Necessariamente, dois caminhos distintos diferem na ordem de um ou mais dos n passos para baixo, dentro dos $n + m$ passos totais.
  - Exemplo, considerando uma grade 2x2:
    - **baixo** - direita - **baixo** - direita (ordens 1 e 3)
    - **baixo** - **baixo** - direita - direita (ordens 1 e 2)

Exemplo: caminho através de uma grade

- **2ª forma de analisar**: novamente, considerando que um caminho é constituído de $n + m$ passos, n para baixo e m para a direita.
  - Dessa forma, podemos escolher n posições dentro das $n + m$ possíveis como sendo passos para baixo. Se tratando de um problema de combinação, já que a ordem de escolha das posições não importam, as combinações (1,3) e (3,1) representam o mesmo caminho (pensando no exemplo anterior)

$$C(n+m, n) = \binom{n+m}{n} = \frac{(n+m)!}{n!\,((\cancel{n}+m)-\cancel{n})!} = \frac{(n+m)!}{n!\,m!}$$

B B

1 2 3 4 5

Esperança matemática

- A Esperança Matemática, o valor esperado ou a expectância de uma variável aleatória nada mais é do que a **média aritmética** de uma variável aleatória.
- Para uma variável aleatória discreta X com valores possíveis $x_1, x_2, x_3, \ldots$ e com as probabilidades representadas pela função $p(x_i)$, o valor esperado calcula-se pela série:

$$E(X) = \sum_{i=1}^{n} x_i . p(x_i)$$

- Sendo que n pode tender ao ∞ se a série for convergente.

Esperança matemática

- Exemplo: uma pequena cirurgia dentária pode ser realizada por um método cujo tempo de recuperação (em dias) é modelado pela variável aleatória X. Determine a Esperança Matemática da variável X da função discreta de probabilidade abaixo

X	5	10	15	20
p_i	0,3	0,2	0,4	0,1

Esperança matemática

- Exemplo: uma pequena cirurgia dentária pode ser realizada por um método cujo tempo de recuperação (em dias) é modelado pela variável aleatória X. Determine a Esperança Matemática da variável X da função discreta de probabilidade abaixo

X	5	10	15	20
p_i	0,3	0,2	0,4	0,1

$$E(X) = 5 * 0{,}3 + 10 * 0{,}2 + 15 * 0{,}4 + 20 * 0{,}1$$
$$E(X) = 11{,}5$$

- Espera-se que o tempo médio de recuperação seja de 11,5 dias

Relação de Recorrência

- Uma relação de recorrência é uma equação definida em termos de si mesma.
- Trata-se de um conceito matemático intimamente ligado ao conceito de **recursão** em computação.
- Por exemplo, qualquer polinômio

$$p_n(x) = c_0 + c_1 x + \ \dots + c_n x^n$$

pode ser escrito **recorrentemente** pela regra de Horn como:

$$p_0(x) = c_n$$
$$p_i(x) = c_{n-i} + x.p_{i-1}(x)$$

p_0 c_n
p_1 $c_{n-1} + c_n X$
p_2 $c_{n+2} + c_{n-1} X + c_n X^2$
$\vdots$
p_n $c_0 + c_1 X \dots c_n X^n$

Relação de Recorrência

- Algo muito importante, especialmente em PC, é que muitas recorrências podem ser expressas simplificadamente através de funções:
  - Isso permite o cálculo analítico de qualquer termo de sequências numéricas recorrentes, independente da quantidade de termos precedentes.
  - Exemplo: i-ésimo termo de uma Progressão Aritmética
    - Relação de recorrência

$$\left.\begin{aligned} a_1 &= X \\ a_n &= a_{n-1} + r \end{aligned}\right\} O(n)$$

    - Termo geral

$$\left.\underline{a_n = a_1 + (n-1)r}\right\} O(1)$$

Relação de Recorrência

- Algo muito importante, especialmente em PC, é que muitas recorrências podem ser expressas simplificadamente através de funções:
  - Isso permite o cálculo analítico de qualquer termo de sequências numéricas recorrentes, independente da quantidade de termos precedentes.
  - Exemplo: Números de Fibonacci
    - Relação de Recorrência

$$\left.\begin{array}{c} F_0 = 0 \text{ e } F_1 = 1 \\ F_n = F_{n-1} + F_{n-2} \end{array}\right\}$$

    - Termo geral

$$F_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$$

Indução Matemática

- Dada uma sequência ou relação de recorrência e valor(es) base conhecido(s), como obter uma expressão fechada para o n-ésimo termo?
- Uma das metodologias matemáticas para solucionar esse tipo de problema é a **indução.**
- Essa metodologia requer uma **hipótese**, usualmente obtida a partir da observação de um conjunto de valores iniciais conhecidos e, possivelmente, ajustes por tentativa-e-erro.

DCo Departamento de Computação

Programação Competitiva unesp

Indução Matemática

- Exemplo:
  - $T_0 = 0$
  - $T_n = 2T_{n-1} + 1$

n	0	1	2	3	4	5	6
T_n	0	1	3	7	15	31	63

- Neste exemplo, uma hipótese fácil de obter é $T_n = 2^n - 1$

Indução Matemática

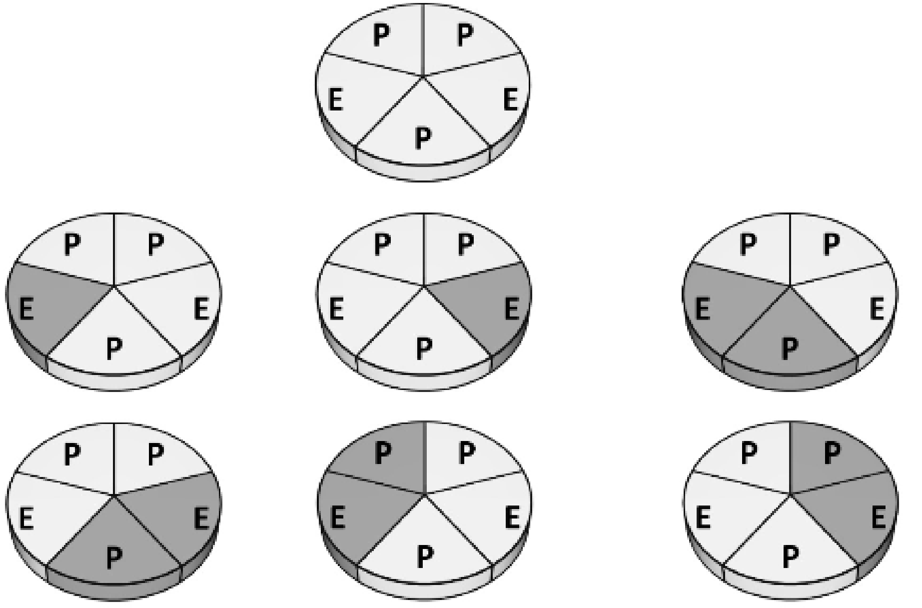
$a_0 \quad a_1 \ldots a_n \quad a_{n+1} \ldots$

- Três passos da Prova por Indução:
  1. Mostre que a hipótese satisfaz o valor base:
     - $T_0 = 2^0 - 1 = 0$
  2. Assuma que a hipótese é válida para qualquer n:
     - $T_n = 2^n - 1$
  3. Use esta hipótese para generalizar para os elementos seguintes:
     - $T_{n+1} = 2T_n + 1 = 2(2^n - 1) + 1 = 2^{n+1} - 2 + 1 = 2^{n+1} - 1$
- As razões da validade deste tipo de prova estão intimamente ligadas às razões do funcionamento de programas recursivos:
  - Garante-se o funcionamento para caso(s) base *(boundary conditions)*
  - Obtém-se o caso geral (*general conditions*) garantindo que este funciona como uma função do caso imediatamente anterior.

Eggfruit Cake (SBC - Nacional 2019)

- Neste problema temos um bolo coberto por frutas, onde cada fruta pode ser uma *eggfruit* (E) ou um *persimmon* (P).
- **Objetivo**: determinar quantas formas há de cortar o bolo em fatias que possuam até S frutas em que pelo menos uma delas é uma fruta do tipo E?

S=2



Eggfruit Cake (SBC - Nacional 2019)

- Primeiramente, já que cada fatia precisa ter pelo menos uma fruta do tipo E, vamos olhar para cada fruta E e ver quantas possibilidades de corte temos que incluam esta posição.
- Exemplo: $S = 3$

P	P	P	E	P	P	P

P	P	P	E	P	P	P
P	P	P	E	P	P	P

P	P	P	E	P	P	P
P	P	P	E	P	P	P
P	P	P	E	P	P	P

Eggfruit Cake (SBC - Nacional 2019)

- Sendo q_E a quantidade de frutas do tipo E, temos que:

$$T = q_E \sum_{i=1}^{S} i = q_E \frac{(1+S)S}{2} = q_E \frac{S+S^2}{2}$$

P	P	P	E	P	P	P

P	P	P	E	P	P	P

P	P	P	E	P	P	P

P	P	P	E	P	P	P

P	P	P	E	P	P	P

P	P	P	E	P	P	P

Eggfruit Cake (SBC - Nacional 2019)

- Mas isso gera um problema: fatias que possuem mais de uma fruta E serão contadas mais de uma vez (1 vez para cada fruta E).

P	P	P	E	P	E	P

P	P	P	E	P	E	P

Eggfruit Cake (SBC - Nacional 2019)

- Mas isso gera um problema: fatias que possuem mais de uma fruta E serão contadas mais de uma vez (1 vez para cada fruta E).
- Solução: para cada fruta E, vamos olhar a sua distância até a fruta anterior (D). Toda fatia de tamanho maior ou igual a D+1 que contenha a fruta anterior deve ser descontada.

P	P	P	E	P	E	P

P	P	P	E	P	E	P

Eggfruit Cake (SBC - Nacional 2019)

- Análise:

Tamanho da fatia	Possibilidades	# de possibilidades
$D + 1$	E…E	1
$D + 2$	XE…E, E…EX	2
$D + 3$	XXE…E, XE…EX, E…EXX	3
…	…	⋮
S	….	S-D

$\underbrace{1+2+3 \ldots (S-D)}$

Eggfruit Cake (SBC - Nacional 2019)

- Valores para subtrair (quantidade de fatias repetidas):

$$R_i = \frac{(1 + S - D_i)(S - D_i)}{2}$$

- Total:

$$A = q_E.\frac{S + S^2}{2} - \sum_i [R_i \mid a\ fruta\ i\ é\ do\ tipo\ E]$$

Desafio: SBC - Regional 2020 – Problema A

- **Problema:** Considere um álbum de figurinha onde todas as figurinhas são iguais. Sendo assim, para completar o álbum basta coletar N figurinhas.
- Porém, quando se compra um pacote de figurinhas, ele contém um número aleatório de figurinhas.
- **Objetivo:** determinar, em média, quantos pacotes são necessários para completar um álbum?

- https://vjudge.net/problem/Gym-102428M

Desafio: SBC - Regional 2020 – Problema A

Entrada

Há apenas uma linha de entrada contendo três inteiros, N, A e B, separados por um espaço, satisfazendo $1 \leq N \leq 10^6$, $0 \leq A \leq B \leq 10^6$ e $B > 0$, onde:

- N é o número de figurinhas necessárias para preencher o álbum;
- A é o número mínimo de figurinhas em um pacote;
- B é o número máximo de figurinhas em um pacote.

O número de figurinhas em cada pacote é um inteiro uniformemente distribuído no intervalo fechado $[A, B]$.

Saída

A saída consiste de apenas uma linha, que deve conter o número esperado de pacotes necessários para completar um álbum. O número será considerado correto se estiver dentro de um erro absoluto ou relativo de 10^{-5} da resposta correta.

Desafio: SBC - Regional 2020 – Problema A

Exemplo de entrada 1	Exemplo de saída 1
40 0 2	40.33333

Exemplo de entrada 2	Exemplo de saída 2
100 1 10	18.72727

Exemplo de entrada 3	Exemplo de saída 3
30 3 3	10.00000

Exemplo de entrada 4	Exemplo de saída 4
314 5 8	48.74556

Desafio: SBC - Regional 2020 – Problema A

- **Dicas:**
  - Pode-se perceber com certa facilidade que este exercício envolve o conceito de esperança matemática. Porém, a aplicação não é tão direta quanto no exemplo que foi dado anteriormente.
  - Um passo importante é buscar determinar uma relação de recorrência x_i tal que $x_i =$ quantidade de pacotes esperado para conseguir $\geq i$ figurinhas.
    - Uma vez determinada essa relação de recorrência, pode ser necessário alguma técnica específica para que ela possa ser calculada de forma eficiente, sem gerar TLE.
  - Caso específico a se considerar: $A = 0$. Não é um caso muito mais complicado que o caso geral, mas é necessário um tratamento especial.
- Resolução: https://www.youtube.com/watch?v=icXLy96-yGI&t=6537s

Referências

Biblioteca de códigos de Thiago Alexandre Domingues de Souza.

Matemática Discreta e Suas Aplicações. Kenneth H. Rosen.

Programming Challenges: The Programming Contest Training Manual. Stevem S. Skiena e Miguel A. Revilla.

https://www.geeksforgeeks.org/permutation-coefficient/

http://wiki.icmc.usp.br/images/a/ac/SCC211Cap6A.pdf

https://mundoeducacao.uol.com.br/matematica/permutacao-envolvendo-elementos-repetidos.htm

https://brasilescola.uol.com.br/matematica/permutacao-com-elementos-repetidos.htm

https://sites.google.com/site/estatisticabasicacc/conteudo/probabilidade/variaveis-aleatorias-discretas/esperanca-matematica