Recorrências e Combinatória

Notas de aula da disciplina
TE: Técnicas de Construção de Algoritmos

Fabiano de Souza Oliveira
(fabiano.oliveira@ime.uerj.br)

Paulo Eustáquio Duarte Pinto
(pauloedp@ime.uerj.br)

julho/2020

Técnicas básicas de Contagem

- Técnicas
- Exemplos

Recorrências

- Formulação
- Solução de Recorrências
- Exemplos

Combinatória como Recorrência

- Combinações Simples/com repetições
- Conjuntos combinatórios especiais
- Exemplos

Exibição de Elementos

Contagem x Exibição de Conjuntos

Técnicas de Contagem:

- Regra do Produto
- Regra da Soma
- Regra da Inclusão/Exclusão
- Princípio da casa de pombo

Expressão da contagem:

- Fórmula combinatória
- Recorrência

Técnicas de Contagem: Regra do Produto

O resultado é a multiplicação de opções, quando a quantidade delas são invariantes umas em relação às outras.

P: Quantas bandeiras distintas de 3 listras podem ser formadas usando 4 cores, tal que não haja listras de mesma cor adjacentes?

R: de 36 maneiras diferentes: $36 = 4.3.3$
(a primeira listra pode ter qualquer uma das 4 cores;
para cada primeira cor, a segunda pode ser 3 outras e,
finalmente, a última tb pode ser escolhida entre 3 opções.

Técnicas de Contagem: Regra da Soma

O resultado é a soma de opções, quando particionam o conjunto de objetos sendo contados.

$$|A \cup B| = |A| + |B| \quad (A \cap B = \emptyset)$$

P: Quantos números no intervalo [100..999] possuem 11 como parte do número?

R: existem 18 números diferentes: 18 = 10+8

(10 números da forma 11x, $0 \leq x \leq 9$
8 números da forma y11, $2 \leq y \leq 9$)

Técnicas Contagem: Regra da Inclusão/Exclusão

O resultado é a união de opções, quando elas se interceptam. $|A \cup B| = |A| + |B| - |A \cap B|$

P: Quantos múltiplos de 3 e de 5 existem no intervalo [1..999]?

R: existem 466 múltiplos diferentes: 466 = 333+199-66

(333 múltiplos de 3 +
199 múltiplos de 5 -
66 múltiplos de 15)

Técnicas Contagem: Princípio da casa de pombo

"Se n+1 pombos são colocados em n gaiolas, então pelo menos uma gaiola deverá conter 2 ou mais pombos".

P: Uma urna tem 4 bolas vermelhas, 7 verdes, 9 azuis e 6 amarelas. Qual o número mínimo de bolas devemos tirar, cegamente, para ter certeza de conseguir 3 bolas de mesma cor?

R: deve-se tirar 9 bolas: 9 = 2.4+1

(generalização: se n gaiolas são ocupadas por nk+1 pombos, então pelo menos uma gaiola conterá k+1 pombos)

Exercícios:

1 - Há 12 moças e 10 rapazes, onde 5 são irmãos (3 moças e 2 rapazes) . Os outros não são parentes. Quantos casamentos são possíveis?

2 - Quantos números distintos podem ser formados com os dígitos 3, 4 e 6, sem repetir dígitos?

3 - Numa festa com 99 convidados, qual o número mínimo de cumprimentos necessários para garantir, independentemente de como os cumprimentos sejam efetuados, que haja um um convidado que tenha sido cumprimentado pelo menos 3 vezes?

4 - Quantos múltiplos de 3 começando com 3 ou 5 existem no intervalo [0..999]?

Fórmulas combinatórias

Permutações, combinações e arranjos simples

$P(n) = n!$

$A(n,p) = n(n-1)\ldots(n-p+1) = n!/(n-p)!$

$C(n,p) = n!/(p!(n-p)!)$

Permutações com repetições - n elementos com repetições:

$r_1, r_2 \ldots r_k \qquad \Sigma\, r_i = n$

$Pr(n) = n!/(r_1! \ldots r_k!)$

Exemplo: Quantas permutações distintas podem ser formadas com os dígitos 3 3 2 2 2?

R: $5!/(2!.3!) = 10$ (22233, 22323, 22332, 23223, 23232, 23322, 32223, 32232, 32322, 33222)

Arranjos com repetições -k elementos distintos tomados n vezes **$Ar(n, k) = k^n$**

Exemplo: Quantos números de 5 bits (0 ou 1) existem?

R: $Ar(5,2) = 2^5$ (00000, 00001, 00010, 00011, ...11101, 11110, 11111)

Combinações com repetições - k elementos distintos tomados n vezes

$Cr(n, k) = C(n+k-1, n)$

Exemplo: Quantos conjuntos distintos de 5 frutas podem ser formados com laranjas, maçãs e peras?

R: $Cr(5,3) = (5+3-1)!/(5!(3-1)!) = 21$ (lllll, llllm, llllp, lllmm, lllmp, lllpp, llmmm, llmmp, llmpp, llppp, lmmmm, lmmmp, lmmpp, lmppp, lpppp, mmmmm, mmmmp, mmmpp, mmppp, mpppp, ppppp)

Exercícios:

5 - De quantas maneiras diferentes pode-se pintar as faces de um cubo usando 2 cores?

6 - Quantas soluções inteiras distintas tem a equação:
$x + y + z = 20$, tal que $x \geq 2$, $y \geq 2$, $z \geq 2$?

Considerações Computacionais -

- **Algoritmos para a computação de fórmulas:**
  - **P(n) , A(n,p):** diretos
  - **C(n,p) , Pr(n), Cr(n,k):** cuidados com os fatoriais no numerador e denominador
  - **Ar(n,k):** cuidado com a ineficiência da exponenciação

- **Inteiros grandes e resultados modulares**

Considerações Computacionais -

- **Combinação**

```
função C(n, p):
    se (p = 0) então
        retornar 1
    senão
        retornar C(n-1,p-1) * n / p
```

Considerações Computacionais -

- **Permutação com Repetição**

função Pr(n, r = [r_1,r_2,..., r_k]):
  se (k = 0) **então**
    retornar P(n)
  senão se r[k] = 0 **então**
    retornar Pr(n, [r_1,r_2,..., r_{k-1}])
  senão
    retornar Pr(n-1,[r_1,r_2,..., r_k-1]) * n / r_k

Considerações Computacionais -

- **Permutação com Repetição**

```
função Ar(n,k):
    se (n = 0) então
        retornar 1
    senão se n é ímpar então
        retornar Ar(n-1,k) * k
    senão
        x ← Ar(n/2,k)
        retornar x*x
```

Considerações Computacionais -

int x: $x \leq 2^{31}-1$ (x $\leq 2^{32}-1$, se **unsigned**)

$x \leq 2{,}1\times10^9$ (x $\leq 4{,}2\times10^9$, se **unsigned**)

$x \leq 12!$ (mesmo se **unsigned**)

long long int x: $x \leq 2^{63}-1$ (x $\leq 2^{64}-1$, se **unsigned**)

$x \leq 9{,}2\times10^{18}$ (x $\leq 18{,}4\times10^{18}$, se **unsigned**)

$x \leq 20!$ (mesmo se **unsigned**)

Considerações Computacionais -

- para valores maiores, usualmente os problemas adotam resultados modulares

  - **somas** e **multiplicações** modulares são mais fáceis

$$(x+y) \bmod n = (x \bmod n + y \bmod n) \bmod n$$
$$(xy) \bmod n = ((x \bmod n)(y \bmod n)) \bmod n$$

  - **divisões** modulares são mais difíceis

$$(x/y) \bmod n \neq ((x \bmod n) / (y \bmod n)) \bmod n$$

Considerações Computacionais -

```
int SomaMod(int a, int b, int n) {
    // retorna (a+b) mod n
    long long int la, lb;
    la = a; lb = b;
    return (la+lb)%n;
}
int MultMod(int a, int b, int n) {
    // retorna (ab) mod n
    long long int la, lb;
    la = a; lb = b;
    return (la*lb)%n;
}
```

Considerações Computacionais -

- **divisão modular**

$(a/b) \bmod n = ((a \bmod n)\ (b' \bmod n)) \bmod n$

onde **b'** é o valor entre **0** e **n-1** tal que

$(b.b') \bmod n = 1$

(**b'** é o ***inverso multiplicativo modular*** de **b**)

MDC(a,b) - Algoritmo de Euclides (Versão recursiva)

```
função MDC (a, b):
    se (b = 0) então
        retornar a
    senão
        retornar MDC(b, a mod b)
```

MDCE(a,b) - Algoritmo de Euclides Estendido (Versão recursiva)

```
função MDCE (a, b):
    //retorna (d,x,y), tal que d é o MDC(a,b) e
    // ax + by = d
    se (b = 0) então
        retornar (a, 1, 0)
    senão
        (d, x', y') ← MDCE(b, a mod b)
        x, y ← y', x' - ⌊a/b⌋.y'
        retornar (d, x, y)
```

MDCE(a,b) - Algoritmo de Euclides Estendido (Versão recursiva)

```
função InvMod (b, n):
    //assume MDC(b,n) = 1
    //retorna b', o inverso mult. de b módulo n
    (d, x, y) ← MDCE(b, n)
    retornar (x+n) mod n
```

Recorrências - São funções nos inteiros expressas fazendo referência a outros valores da função.

Exs: $T(n) = T(n-1) + 1$ $\quad T(0) = 0$
$T(n) = 2.T(n-1) + 1$ $\quad T(0) = 0$
$T(n) = n.T(n-1)$ $\quad T(0) = 1$

Forma Fechada de uma Recorrência- Função que não emprega referência a outros valores da função, mas uma "fórmula", e equivale à recorrência

Ex: $T(n) = T(n-1) + n$ $\quad T(0) = 0$
$T'(n) = n(n+1)/2$
$T'(n)$ é a forma fechada de $T(n)$

Solução de recorrências:

Obter a Forma Fechada:

a) Recorrência clássica
b) Indução finita
c) Equações diferenciais

Complexidade:

O(1), se for uma fórmula simples
O(log n), se for uma exponencial
O(n), se for polinômio de grau n

Considerações Computacionais -

- **Avaliação de Polinômios $f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$**

```
função Aval(a = [a0, a1, … an], n, x):
    i ← 0;    b ← 1
    r ← a[i]
    // invariante: b = x^i    ,    r = a_i x^i + a_{i-1} x^{i-1} + … + a_0
    enquanto i ≠ n faça
        b ← b * x;   i ← i+1
        r ← r + a[i] * b
    retornar r
```

Recorrências clássicas

a) Soma dos n primeiros inteiros positivos

$T(n) = T(n-1) + n; \quad T(0) = 0;$

$T(n) = n(n+1)/2$

b) Produto dos n primeiros inteiros positivos

$T(n) = n.T(n-1) \quad T(0) = 1;$

$T(n) = n!$

c) Torre de Hanoi

$T(n) = 2.T(n-1) + 1 \quad T(0) = 0;$

$T(n) = 2^n - 1$

Recorrências clássicas

d) Série de Fibonacci

$T(n) = T(n-1) + T(n-2) \qquad T(0) = 0, \;\; T(1) = 1$

$T(n) = (A^n - B^n)/\ 5^{1/2}\ ; \quad A = (1 + 5^{1/2})/2; \;\; B = (1 - 5^{1/2})/2$

e) Recorrência de Catalão

$T(n) = \;\Sigma T(i).T(n-i-1),\ 0 \leq i \leq n-1, \;\; T(0) = 1$

$T(n) = C(2n,n)/(n+1)$

f) Recorrências Polinomiais

$T(n) = T(n-1) + P(n,k); \qquad T(1) = c;$

$T(n) = P(n,\ k+1)$

(onde $P(n,k)$ denota algum polinômio em n de grau k)

- $T(n) = a_k n^k + a_{k-1} n^{k-1} + \dots + a_0$

  $T(0) = t_0$; $T(1) = t_1$; ... ; $T(k) = t_k$;

  $t_0 = a_k 0^k + a_{k-1} 0^{k-1} + \dots + a_0$
  $t_1 = a_k 1^k + a_{k-1} 1^{k-1} + \dots + a_0$

  ...
  $t_k = a_k k^k + a_{k-1} k^{k-1} + \dots + a_0$

Solução de recorrências:

Sem a Forma Fechada:

a) Recursão (com Memorização):

Complexidade: $O(n\ r(n))$, onde $r(n)$ é o número de referências à própria função

Casos especiais: $O(n)$ se $r(n) = O(1)$

$O(n^2)$ se $r(n) = O(n)$

b) Recorrência Linear:

Complexidade: $O(\log n\ r(n)^3)$

Caso especial: $O(\log n)$ se $r(n) = O(1)$

Considerações Computacionais -

- **Recorrência Linear:**
  $T(n) = a_1T(n-1) + a_2T(n-2) + \ldots + a_kT(n-k)$
  $T(1) = t_1;\ T(2) = t_2;\ \ldots;\ T(k) = t_k;$

$$\begin{bmatrix} a_1 & a_2 & \ldots & a_{k-1} & a_k \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} t_{n-1} \\ t_{n-2} \\ \ldots \\ t_{n-k+1} \\ t_{n-k} \end{bmatrix} = \begin{bmatrix} t_n \\ t_{n-1} \\ \ldots \\ t_{n-k+2} \\ t_{n-k+1} \end{bmatrix}$$

Considerações Computacionais -

- **Recorrência Linear:**
  $T(n) = a_1 T(n-1) + a_2 T(n-2) + \ldots + a_k T(n-k)$
  $T(1) = t_1;\ T(2) = t_2;\ \ldots;\ T(k) = t_k;$

$$\begin{bmatrix} a_1 & a_2 & \ldots & a_{k-1} & a_k \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} t_{n-2} \\ t_{n-3} \\ \ldots \\ t_{n-k} \\ t_{n-k-1} \end{bmatrix} = \begin{bmatrix} t_{n-1} \\ t_{n-2} \\ \ldots \\ t_{n-k+1} \\ t_{n-k} \end{bmatrix}$$

Considerações Computacionais -

- **Recorrência Linear:**
  $T(n) = a_1T(n-1) + a_2T(n-2) + \dots + a_kT(n-k)$
  $T(1) = t_1;\ T(2) = t_2;\ \dots;\ T(k) = t_k;$

$$\begin{bmatrix} a_1 & a_2 & \dots & a_{k-1} & a_k \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}^2 \times \begin{bmatrix} t_{n-2} \\ t_{n-3} \\ \dots \\ t_{n-k} \\ t_{n-k-1} \end{bmatrix} = \begin{bmatrix} t_n \\ t_{n-1} \\ \dots \\ t_{n-k+2} \\ t_{n-k+1} \end{bmatrix}$$

Considerações Computacionais -

- **Recorrência Linear:**
  $T(n) = a_1T(n-1) + a_2T(n-2) + \dots + a_kT(n-k)$
  $T(1) = t_1;\ T(2) = t_2;\ \dots;\ T(k) = t_k$

$$\begin{bmatrix} a_1 & a_2 & \dots & a_{k-1} & a_k \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}^{n-k} \times \begin{bmatrix} t_k \\ t_{k-1} \\ \dots \\ t_2 \\ t_1 \end{bmatrix} = \begin{bmatrix} t_n \\ t_{n-1} \\ \dots \\ t_{n-k+2} \\ t_{n-k+1} \end{bmatrix}$$

Considerações Computacionais -

- **Exponenciação eficiente de matrizes**

função Exp(A, n):
  // assume: A é matriz quadrada de dimensão k
  // retorna A^n
  se (n = 0) **então**
    retornar I_k (matriz identidade de dimensão k)
  senão se n é ímpar **então**
    retornar Exp(A, n-1) × A
  senão
    X ← Exp(A, n/2)
    retornar X × X

Considerações Computacionais -

- **Exponenciação eficiente de matrizes**

```
função MultiplicaMatrizes(A, B):
   // assume: A e B são matrizes quadradas de dimensão k
   // retorna A×B
   R ← 0_k (matriz quadrada nula de dimensão k)
   para i ← 1 até k faça
      para j ← 1 até k faça
         para z ← 1 até k faça
            R[i,j] ← R[i,j] + A[i,z]*B[z,j]
   retornar R
```

Considerações Computacionais -

- **Ex: Cálculo do n-ésimo termo de Fibonacci em O(log n)**

Fib(n) = Fib(n-1) + Fib(n-2); Fib(1) = 1; Fib(2) = 1

função Fib(n):

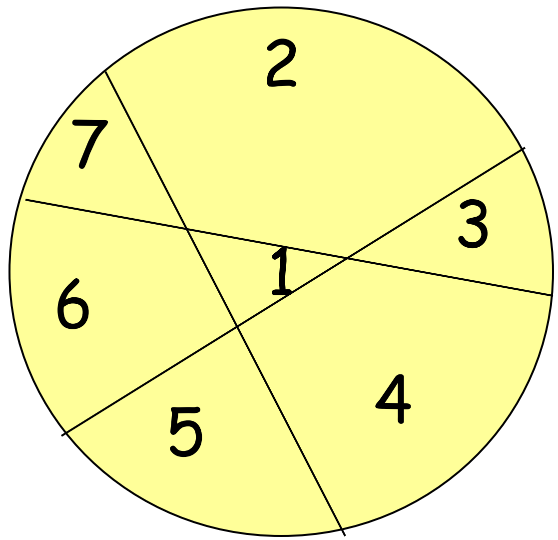
// k = 2; $a_1 = a_2 = \quad t_1 = t_2 = 1$

$M \leftarrow \begin{bmatrix} a_1 & a_2 \\ 1 & 0 \end{bmatrix}$

$R \leftarrow Exp(M, n-2)$

retornar $R[1,1] * t_1 + R[1,2] * t_2$

Pr. 1 – Pizza Cutting (10079)



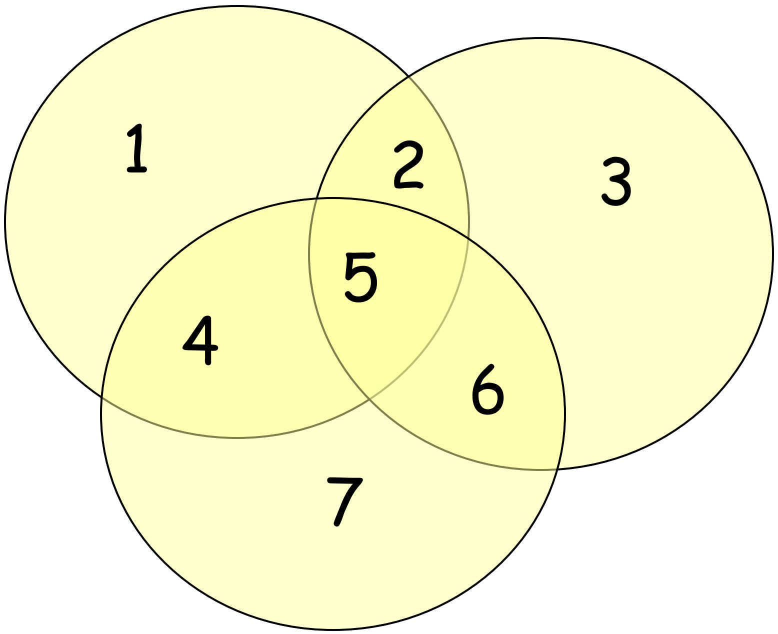
Qual o maior número de pedaços com n cortes?

$T(0) = 1$ $T(1) = 2;$ $T(2) = 4;$ $T(n) = 2^n$?

R: Não.

$T(n) = T(n-1) + n$ => $T(n) = ?$

Pr. 2 - Interseção de círculos



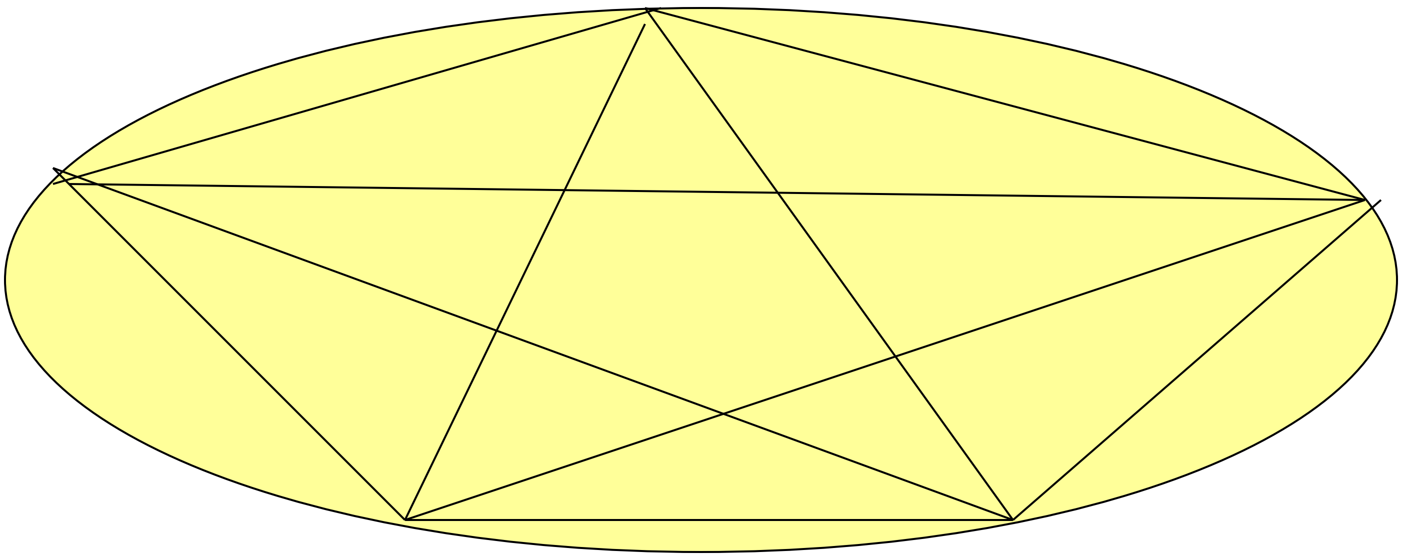
Qual o maior número de pedaços com n círculos?

$T(1) = 1$; $T(2) = 3$; $T(3) = 7$; $T(n) = 2^n - 1$?

R: Não.

$T(n) = T(n-1) + 2(n-1)$ => $T(n) = n(n-1)+1$

Pr. 3 - How many pieces of Land? (10213)



Maior número de pedaços com polígono de n vértices, cuidadosamente colocados?

$T(0) = 1$ $T(1) = 1$; $T(2) = 2$; $T(3) = 4$

$T(4) = 8$ $T(5) = 16$ $T(n) = 2^{n-1}$?

R: Não.

$T(n) = T(n-1) + \Sigma\ \{\ (i-1)(n-1-i) + 1\ :\ 1 \leq i \leq n-1\ \}$

$= T(n-1) + (n-1) + \Sigma\ \{\ i\ (n-2-i)\ :\ 0 \leq i \leq n-2\ \}$

$= T(n-1) + (n-1) + \Sigma\ \{\ i\ (n-2-i)\ :\ 1 \leq i \leq n-3\ \}$

$= T(n-1) + (n-1) + n\ \Sigma\ \{\ i\ :\ 1 \leq i \leq n-3\ \}$

$-\ \Sigma\ \{\ 2\ :\ 1 \leq i \leq n-3\ \}$

$-\ \Sigma\ \{\ i^2\ :\ 1 \leq i \leq n-3\ \}$

$(= T(n-1) + P(n,3))$

$T(n) = n\ (n-1)(n^2 - 5n + 18)/24 + 1$

$\Rightarrow\ T(6) = 31\ (!)$

Exercício 7: How many points of Intersection? (10790)

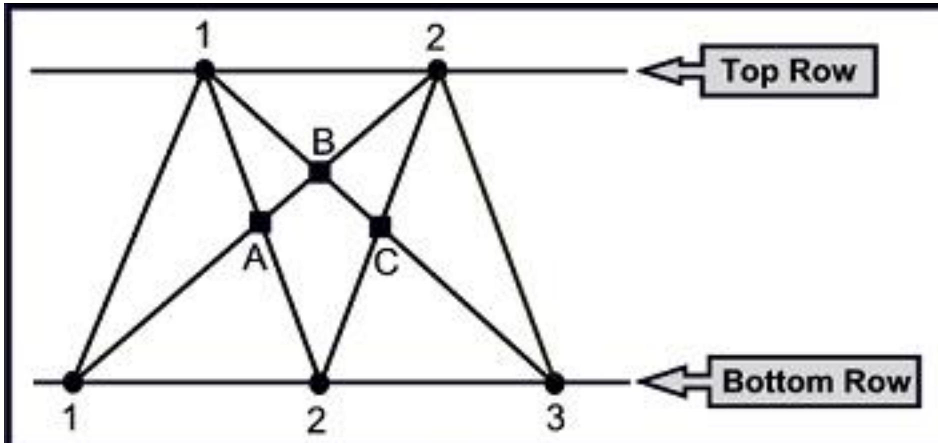


Figure: 2 dots on the top row and 3 dots on the bottom row. And total no of internal intersections is P(2,3)=3. In this picture these intersections are marked as A, B and C.

Dadas duas retas paralelas, **3** pontos na reta superior e **4** na inferior, qual o número máximo de pontos de intersecção? E genericamente?

Pr. 4: Partição de Inteiros:

dado n, quantos conjuntos distintos de inteiros positivos existem tal que sua soma seja n?

Exemplo: Temos $T(5) = 7$. Os conjuntos são:
{5}, {4, 1},{3, 2},{3, 1, 1},{2, 2, 1},{2, 1, 1, 1},{1, 1, 1, 1, 1}

Recorrência para T(n), ou melhor T(n, n), onde

T(n, p) = quantidade de conjuntos distintos de inteiros positivos com soma igual a n e maior elemento p

Idéia: separar os conjuntos de acordo com o maior elemento.

$$T(n, p) = T(n-p, p) + T(n, p-1)$$
$$T(n, p) = 0, \text{ se } (n < 0 \text{ ou } p = 0)$$
$$T(0, p) = 1, p > 0$$

Partição de Inteiros: cálculo de T(5, 5)

(n) \ (p)	0	1	2	3	4	5
0	0	1	1	1	1	1
1	0	1	1	1	1	1
2	0	1	2	2	2	2
3	0	1	2	3	3	3
4	0	1	3	4	5	5
5	0	1	3	5	6	7

Pr. 5: Partição de Conjuntos:

dado um conjunto com n elementos, de quantas maneiras distintas pode ser particionado em p subconjuntos?

Exemplo: Temos $T(4, 3) = 6$. As partições são:
({1},{2},{3,4}), ({1},{3},{2,4}), ({1},{4},{2,3}), ({2},{3},{1,4}), ({2},{4},{1,3}), ({3},{4},{1,2})

Recorrência para T(n, p):

Idéia: Calcular a formação a partir dos conjuntos para (n-1) elementos.

$$T(n, p) = T(n-1, p).p + T(n-1, p-1), \qquad p < n$$
$$T(n, 1) = 1$$
$$T(n, n) = 1$$

Partição de conjuntos: cálculo de T(5, 5)

	1	2	3	4	5
1	1	-	-	-	-
2	1	1	-	-	-
3	1	3	1	-	-
4	1	7	6	1	-
5	1	15	25	10	1

1. Combinações

$$C(n, p) = C(n-1, p).n/(n-p)$$
$$C(p, p) = 1,$$

ou

$$C(n, p) = C(n, p-1)(n-p+1)/p$$
$$C(n, 0) = 1,$$

2. Triângulo de Pascal

$$C(n, p) = C(n-1, p) + C(n-1, p-1)$$
$$C(n, 0) = 1$$
$$C(n, p) = 0, \text{ se } (p > n)$$

Triângulo de Pascal

	0	1	2	3	4	5	6	7
0	1	-	-	-	-	-	-	-
1	1	1	-	-	-	-	-	-
2	1	2	1	-	-	-	-	-
3	1	3	3	1	-	-	-	-
4	1	4	6	4	1	-	-	-
5	1	5	10	10	5	1	-	-
6	1	6	15	20	15	6	1	-
7	1	7	21	35	35	21	7	1

2. Algoritmo para o Triângulo de Pascal

$$C(n, p) = C(n-1, p) + C(n-1, p-1)$$

$$C(n, 0) = 1$$

$$C(n, p) = 0, \text{ se } (p > n)$$

```
Triângulo(n);
    C[0,0] ← 1
    Para j ← 1 até n incl.:
        C[0,j] ← 0
    Para i ← 1 até n incl.:
        C[i,0] ← 1
        Para j ← 1 até i incl.:
            C[i,j] ← C[i-1,j]+C[i-1,j-1]
        Para j ← i+1 até n incl.:
            C[i,j] ← 0
```

Triângulo de Pascal - Exercício para casa

Mostrar que a soma das diagonais invertidas são números de Fibonacci:

	0	1	2	3	4	5	6	7
0	1	-	-	-	-	-	-	-
1	1	1	-	-	-	-	-	-
2	1	2	1	-	-	-	-	-
3	1	3	3	1	-	-	-	-
4	1	4	6	4	1	-	-	-
5	1	5	10	10	5	1	-	-
6	1	6	*15*	*20*	15	6	1	-
7	1	7	21	35	35	21	7	1

2

3

5

8

13

3. Números de Euler: contam o número de permutações com dado número de "runs"

$$E(n, p) = E(n-1, p).p + E(n-1, p-1).(n-p+1)$$
$$E(n, 0) = 0$$
$$E(0, 1) = 1 \text{ (artificial)}$$

Exemplo: E(4, 2)

$E(3, 2) = 3$ (132, 231, 312) $E(3, 1) = 1$ (123);

$E(4, 2) = 3.2 + 1.3 = 9$

(1342, 1324, 2341, 2314, 3412, 3124, 4123, 1423, 1243)

4. Números de Stirling: contam o número de permutações com dado número de ciclos

$$S(n, p) = S(n-1, p).(n-1) + S(n-1, p-1)$$
$$S(0, 0) = 1 \text{ (artificial)}$$

Exemplo: S(4, 2)

$S(3, 2) = 3$ (132, 321, 213) $S(3, 1) = 2$ (231, 312,);

$S(4, 2) = 3.3 + 1 = 11$

(4321, 1423, 1342, 4213, 3412, 3241, 4132, 2431, 2143, 2314, 3124)

5. Desarranjos.

São permutações de 1..n onde nenhum elemento é igual ao próprio índice

D(n) = número de desarranjos de n

n	1	2	3	4	5	6	7
D(n)	0	1	2	9	44	256	1854

Exemplos de desarranjos:

n = 2. 1 desarranjo: (2,1)

n = 3. 2 desarranjos: (2,3,1) (3,2,1):

n = 4. 9 desarranjos: (2,1,4,3), (2,3,4,1), (2,4,1,3), (3,1,4,2), (3,4,1,2), (3,4,2,1), (4,1,2,3), (4,3,1,2) , (4,3,2,1)

5. Desarranjos.

Recorrência para calcular D(n):

$D(1) = 0;$
$D(2) = 1;$
$D(n) = (n-1)(D(n-1)+D(n-2)).$

Confirmação:

$D(3) = 2(0+1) = 2;$ $D(4) = 3(1+2) = 9;$ $D(5) = 4(2+9) = 44$

Solução da recorrência:

$D(n) =$ $!n$ (Subfatorial de n) $= n!\sum^{n}_{k=0} (-1)^k/k!$

Confirmação:

$D(4) = 4!(1-1+1/2!-1/3!+1/4!) = 4!/4!(24-24+12-4+1) = 9$

6. Sequências de 1's. World Cup Noise (10450)

Dado n, quantos números binários T(n) com n bits não contém 2 ou mais bits 1 em sequência?

Ex: T(3) = 5 (000, 100, 010, 001, 101)

Recorrência:

2 casos, pois o final do número é 0 ou 01

$T(1) = 2; \quad T(2) = 3$

$T(n) = T(n-1) + T(n-2), \quad \text{se } (n > 2)$

$T(n) = FIB(n+2)$!!!

7. Agrupamentos. n Group k (10568)

Quantos agrupamentos distintos T(n, k) existem para n elementos havendo k elementos por grupo, exceto 1 deles que contém os restantes, quando n mod k ≠ 0.

Exemplo: T(5, 2) = 15, correspondendo a:

A	BC	DE	AC	BE	D
A	BD	CE	AD	B	CE
A	BE	CD	AD	BC	E
AB	C	DE	AD	BE	C
AB	CD	E	AE	B	CD
AB	CE	D	AE	BC	D
AC	B	DE	AE	BD	C
AC	BD	E			

Exercícios:

8 - Quantos agrupamentos de 6 elementos podem ser formados com 8 pessoas? E com 12?

Agrupamentos. **n Group k (10568)**

A B C D E $\quad T(5, 2) = 15$

Recorrência: dividir em 2 casos

Seja $q = n \bmod k$

$T(n, k) = 1,\qquad$ se $(0 \le n \le k)$

$T(n, k) = C(n-1, k-1).T(n-k, k),\;$ se $(n > k)$ e $(q = 0)$

$T(n, k) = C(n-1, k-1).T(n-k, k) + C(n-1, q-1).T(n-q, k)$,
se $(n > k)$ e $(q \neq 0)$

Agrupamentos. n Group k (10568)

Seja $q = n \bmod k$

$T(n, k) = 1,$ se $(n \leq q)$ e $(n \geq 0)$

$T(n, k) = C(n-1, k-1).T(n-k, k),$ se $(n > k)$ e $(q = 0)$

$T(n, k) = C(n-1, k-1).T(n-k, k) + C(n-1, q-1).T(n-q, k)$,

se $(n > k)$ e $(q \neq 0)$

Exemplo: $T(7,4) = C(6,3).T(3,4) + C(6,2).T(4,4) = 20*1+15*1=35$

$T(10,3) = C(9,2).T(7,3) + C(9,0).T(9,3) = 2520+280=2800$

	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
2	1	1	3	3	15	15	105	105	945	945
3	1	1	1	4	10	10	70	280	280	2800
4	1	1	1	1	5	15	35	35	315	1575
5	1	1	1	1	1	6	21	56	126	126

8. Combinações com elementos repetidos.

Combinations, Once Again (10532)

k elementos com repetições: $r_1, r_2 \ldots r_k$, $\Sigma\, r_i = n$

Quer-se saber Cr(n, p)

Ex: Dados 1 1 2 2 2 2 4 5 5 5 5 5 8 8, calcular Cr(14, 5)

Recorrência: trabalha-se com a recorrência T(p,q), que significa de quantas maneiras pode-se organizar os p tipos iniciais de elementos usando q elementos no total.

$T(p,0) = 1;$

$T(0,q) = 0,\ q > 0$

$T(p,q) = \Sigma_{i=0}^{\min(rp,q)} Cr(p-1,q-i)$

$T(p,q) = 0$, nos demais casos.

8. Combinações com elementos repetidos.

Combinations, Once Again (10532) - Cr(14, 5)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1(2)	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
2(4)	1	2	3	3	3	2	1	0	0	0	0	0	0	0	0
3(1)	1	3	5	6	6	5	3	1	0	0	0	0	0	0	0
4(5)	1	4	9	15	21	26	28	26	21	15	9	4	1	0	0
5(2)	1	5	14	28	45	62	75	80	75	62	43	27	14	5	1

8. Combinações com (q) elementos repetidos

$T(p,0) = 1;$

$T(0,q) = 0,\ q > 0$

$T(p,q) = \Sigma_{i=0}^{\min(rp,q)} Cr(p-1,q-i)$

$T(p,q) = 0$, nos demais casos.

k elementos com repetições: $r_1, r_2 \ldots r_k, \Sigma\, r_i = n$

```
Combrep;
    T[0,0] ← 1
    Para j ← 1 até n incl.:
        T[0,j] ← 0
    Para i ← 1 até k incl.:
        T[i,0] ← 1
        m ← min(r_i,q)
        Para j ← 1 até q incl.:
            ma ← min (m,j)
            Para k ←  0 até ma incl.:
                T[i,j] ← T[i,j]+T[i-1,j-k]
```

Exercício 9: Calcular Cr(8,3) para a situação:

n	1	2	3	4	5	6	7	8
	1	2	2	3	3	3	4	4

Exercícios:

10 - Dadas 5 letras e 6 espaços em branco, de quantas maneiras diferentes se pode formatar o string de 11 caracteres correspondente, sem que 2 letras estejam em sequência?

9. Formatação. Stripes (10541)

Dados p objetos e n espaços consecutivos, quantas distribuições $T(n, p)$ distintas existem tal que nunca haja dois objetos vizinhos?

Ex: $T(5, 2) = 6$ (10100, 10010, 10001, 01010, 01001, 00101)

Recorrência: 2 casos, considerando se a primeira posição do string é branco ou não.

$T(n, 0) = 1$; $T(1, 1) = 1$;

$T(n, p) = 0$, se $(n < 2p-1)$

$T(n, p) = T(n-1, p) + T(n-2, p-1)$, c.c

9. Formatação. Stripes (10541)

$T(n, 0) = 1; \quad T(1, 1) = 1;$

$T(n, p) = 0, \ se\ (n < 2p-1)$

$T(n, p) = T(n-1, p) + T(n-2, p-1), \ c.c$

$$T(n,p) = C(n-p-1, p) + C(n-p-1, p-2) + 2C(n-p-1, p-1)$$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	0	0	0	1	3	6	10	15	21	28	36	45	55	66	78
3	0	0	0	0	0	1	4	10	20	35	56	84	120	165	210

FIM