Contents

1 Identities

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\sum_{i=1}^{n} i^3 = \frac{n^2(n+1)^2}{4} = \left(\sum_{i=1}^{n} i\right)^2$$

2 Number Theory

2.1 Fundamental theorem of arithmetic

Every integer greater than 1 can be represented uniquely as a product of prime numbers, up to the order of the factors.

$$n = p_1^{\alpha_1} p_2^{\alpha_2} ... p_k^{\alpha_k} \tag{1}$$

2.1.1 LCM and GCD

$$\begin{aligned}
a &= p_1^{\alpha_1} p_2^{\alpha_2} ... p_k^{\alpha_k} \\
b &= p_1^{\beta_1} p_2^{\beta_2} ... p_k^{\beta_k} \\
(a,b) &= p_1^{\min \alpha_1,\beta_1} p_2^{\min \alpha_2,\beta_2} ... p_k^{\min \alpha_k,\beta_k} \\
[a,b] &= p_1^{\max \alpha_1,\beta_1} p_2^{\max \alpha_2,\beta_2} ... p_k^{\max \alpha_k,\beta_k}
\end{aligned} \tag{2}$$

2.2 Fermat's Theorems

Let P be a prime number and a an integer, then:

$$a^p \equiv a \pmod{p}$$

$$a^{p-1} \equiv 1 \pmod{p}$$

Lemma: Let p be a prime number and a and b integers, then:

$$(a+b)^p \equiv a^p + b^p \pmod{p}$$

Lemma: Let p be a prime number and a an integer. The inverse of a modulo p is a^{p-2}:

$$a^{-1} \equiv a^{p-2} \pmod{p}$$

2.3 Goldbach's Conjecture

"Todo número par, maior que 2, pode se ser escrito como a soma de dois primos. ". Válido pra todo número até $4 \cdot 10^{18}$, mas não tem prova.

Para os números ímpares, sempre pode ser escrito como soma de 3 números primos, tira 3, e pega os dois que formam o número par que restar, porém caso o número primo - 2 seja primo, daí ele consegue ser escrito como 2 também.

2.4 Taking modulo at the exponent

Se a e m são coprimos entre si então:

$$a^m \equiv a^{n \bmod \varphi(m)} \pmod{m} \tag{3}$$

3 Geometry

3.1 Convex and Concave Polygons

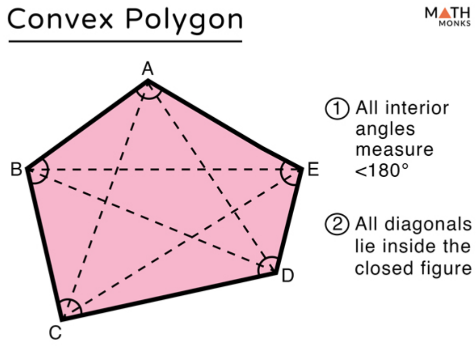


Figure 1: Convex Polygon

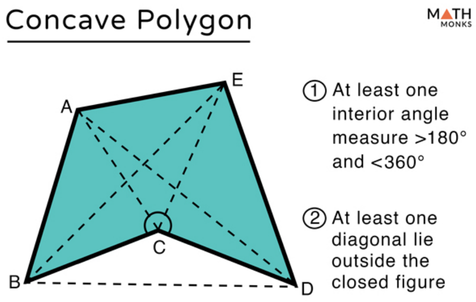
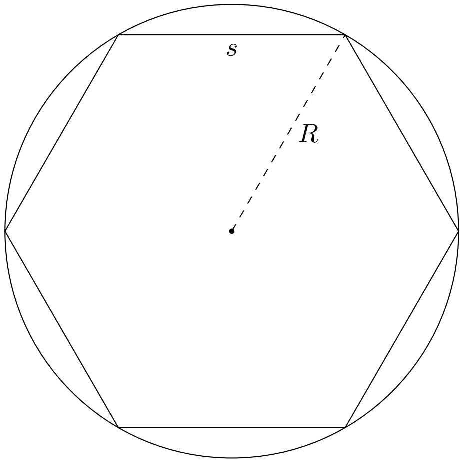


Figure 2: Concave Polygon

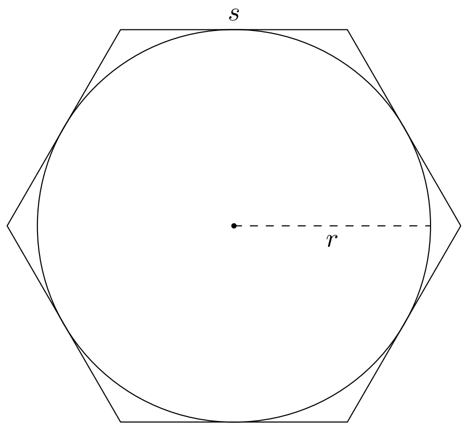
Figure 3: Two Types of Polygons

3.2 Regular polygon circumradius



$$R = \frac{s}{2}\csc\frac{\pi}{n}$$

3.3 Regular polygon inscribed circle radius



$$r = R\cos\frac{\pi}{n}$$

3.4 Area of regular polygons

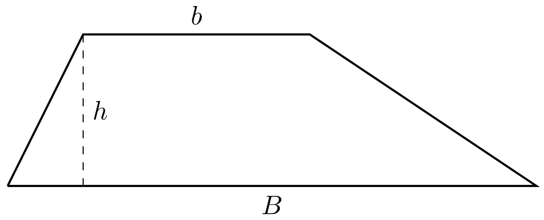
- Let n be the number of sides of the regular polygon, the area can be found using one of the values below:

  1. the lenght of one of the sides (s)
  2. apothem, the radius of the inscribed circle (r)
  3. the radius of the circumscribed circle (R)

$$A = \frac{1}{2}nrs = \frac{1}{4}ns^2\cot\frac{\pi}{n} = nr^2\tan\frac{\pi}{n} = \frac{1}{2}nR^2\sin\frac{2\pi}{n}$$

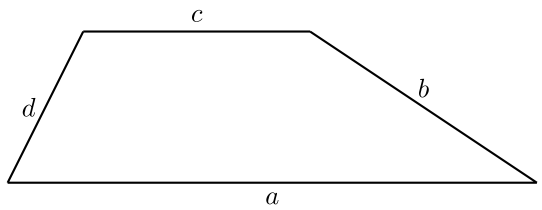
3.5 Trapezium

3.5.1 Area, knowing base and height



$$A = \frac{(B+b)h}{2}$$

3.5.2 Area knowing only sides



$$e = \frac{d^2 - b^2 + a^2 - 2ac + c^2}{2a - 2c}$$

$$h = \sqrt{d^2 - e^2}$$

$$A = \frac{h(a+c)}{2}$$

3.6 Dot product

The dot product of vectors **u** and **v** in n dimensions is given by:

$$\langle \mathbf{u}, \mathbf{v} \rangle = u_1 \cdot v_1 + u_2 \cdot v_2 + \ldots + u_n \cdot v_n$$

3.7 Magnitude

The magnitude of a vector **v** in n dimensions is given by:

$$|\mathbf{v}| = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

The dot product of two Euclidean vectors **u** and **v** is defined by

$$\langle \mathbf{u}, \mathbf{v} \rangle = |\mathbf{u}| \cdot |\mathbf{v}| \cdot \cos(\theta)$$

3.8 Equação reduzida da reta

$$y = mx + b \tag{4}$$

Onde m é o coeficiente angular que representa a taxa de variação da reta: consiste no número de unidades que y varia para cada unidade de variação de x no sentido positivo do eixo horizontal. O coeficiente linear b é o valor no qual a reta intercepta o eixo y.
Não pode representar retas verticais

3.9 Equação geral da reta

$$ax + by + c = 0 \tag{5}$$

3.10 Equação geral da reta a partir de dois pontos

Seja $P = (x_p, y_p)$ e $Q = (x_p, y_p)$.
$a = y_p - y_q$, $b = x_q - x_p$, $c = x_p y_q - x_q y_p$

3.11 Inclinação da reta a partir de dois pontos

Seja $P = (x_p, y_p)$ e $Q = (x_p, y_p)$, a inclinação da reta é dada por:

$$m = \frac{y_q - y_p}{x_q - x_p} \tag{6}$$

3.12 Pertecimento de ponto a reta

Seja r uma reta com equação geral $ax + by + c = 0$ e $P = (x_p, y_p)$ um ponto qualquer, $P \in r$ se, e somente se : $ax_p + bx_p + c = 0$

3.13 Distância entre dois pontos (euclidiana)

A distância euclidiana entre dois pontos $A = (x_a, y_a)$ e $B = (x_b, y_b)$ é dada por :

$$\sqrt{(x_a - y_a)^2 + (y_a - y_b)^2}$$

3.14 Distância entre ponto e reta

A distância de um ponto P a uma reta r é definida como a menor disância possível entre todos os pontos de r e P. A menor distância será aquela entre o ponto P e o ponto de intersecção Q de r com a reta perpendicular a r que passa por P.
A distância d entre $P = (x_p, y_p)$ e a reta $ax + by + c = 0$ é dada por :

$$\frac{|ax_p + by_p + c|}{\sqrt{a^2 + b^2}} \tag{7}$$

As coordenadas do ponto Q = (x_q, y_q) sáo dadas por :

$$x_q = \frac{b(bx_p - ay_p) - ac}{a^2 + b^2} \quad , \quad y_q = \frac{a(-bx_p + ay_p) - bc}{a^2 + b^2} \tag{8}$$

4 Probability

4.1 Expected Value

The expected value for X is the probability of X occurs multiplied by the quantification of X
$E(X) = \sum P(i) * X(i)$

4.2 Linearity of EV

$E(X + Y) = E(X) + E(Y)$

5 Polynomial

5.1 Bhaskara

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{9}$$

5.2 Pascal's Triangle

5.3 N-th first terms of P-th column in Pascal Triangle

$$\binom{p}{p} + \binom{p+1}{p} + ... + \binom{p+n}{p} = \binom{p+n+1}{p+1} \tag{10}$$

5.4 Number of odd numbers in the N-th line of pascal triangle

There is a mathematical relation which gives the number of odd numbers in the N-th row of pascal's triangle. The theorem states that the number of odd numbers in N-th row is equal to 2 raised to the number of ones in the binary representation of N.

6 Graphs

6.0.1 Bipartite Graph

Bipartite Graph is a special graph with the following characteristics: the set of vertices V can be partitioned into two disjoint sets V1 and V2 and all undirected edges (u, v) $\in$ E have the property that u $\in$ V1 and v $\in$ V2 . This makes a Bipartite Graph free from odd-length cycle. Not that by this definition is possible to have isolated vertices.

6.1 2-SAT

In order for a 2-SAT problem to have a solution, it is necessary and sufficient that for any variable x the vertices x and $\neg x$ are in different

strongly connected components of the strong connection of the implication graph.
This criterion can be verified in $O(n+m)$ time by finding all strongly connected components.

6.2 Topological Sorting

Definition: You are given a **directed** graph with n vertices and m edges. You have to find an order of the vertices, so that every edge leads from the vertex with a smaller index to a vertex with a larger one.
Topological order can be non-unique !
A Topological order **may not exist** at all. It only exists, if the directed graph **contains no cycles**. Otherwise because there is a contradiction: if there is a cycle containing the vertices a and b, then a needs to have a smaller index than b (since you can reach b from a) and also a bigger one (as you can reach a from b). Every acyclic directed graph contains at least one topological order.

6.3 Strongly Connected Components

You are given a directed graph G with vertices V and edges E. It is possible that there are loops and multiple edges. Let's denote n as number of vertices and m as number of edges in G.
Strongly connected component is a maximal subset of vertices C such that any two vertices of this subset are reachable from each other, i.e. for any $u, v \in C$:

$$u \mapsto v, v \mapsto u$$

where $\mapsto$ means reachability, i.e. existence of the path from first vertex to the second.
The most important property of the condensation graph is that it is a **DAG**. Indeed, suppose that there is an edge between C and C', let's prove that there is no edge from C' to C. Suppose that $C' \mapsto C$. Then there are two vertices $u' \in C$ and $v' \in C'$ such that $v' \mapsto u'$. But since u and u' are in the same strongly connected component then there is a path between them; the same for v and v'. As a result, if we join these paths we have that $v \mapsto u$ and at the same time $u \mapsto v$. Therefore u and v should be at the same strongly connected component, so this is contradiction. This completes the proof.

6.4 Minimum spanning tree

6.4.1 Propeties

- A minimum spanning tree of a graph is unique, if the weight of all the edges are distinct. Otherwise, there may be multiple minimum spanning trees. (Specific algorithms typically output one of the possible minimum spanning trees).

- Minimum spanning tree is also the tree with minimum product of weights of edges. (It can be easily proved by replacing the weights of all edges with their logarithms)

6.5 Eulerian path

A Eulerian path is a path in a graph that passes through all of its edges exactly once. A Eulerian cycle is a Eulerian path that is a cycle.
An Eulerian cycle exists if and only if the degrees of all vertices are even
And an Eulerian path exists if and only if the number of vertices with odd degrees is two (or zero, in the case of the existence of a Eulerian cycle).

6.6 Network

A **network**is a directed graph G with vertice V and edges E combined with a function c, which assigns each edge $e \in E$ a non-negative integer value, the $capacity$ of e.

6.7 Flow network

Is a **network** with two vertices labeled as **source** and **sink**.

6.7.1 Propeties

- The flow of an edge cannot exceed the capacity

$$f(e) <= c(e)$$

- And the sum of the incoming flow of a vertex u has to be equal to the sum of the outgoing flow of u except in the source and sink vertices.
- The source vertex s only has an outgoing flow, and the sink vertex t has only incoming flow.

6.8 Prufer Code

The Prüfer code is a way of encoding a labeled tree with n vertices using a sequence of $n-2$ integers in the interval $[0; n-1]$. This encoding also acts as a bijection between all spanning trees of a complete graph and the numerical sequences.
The Prüfer code is constructed as follows. We will repeat the following procedure $n-2$ times: we select the leaf of the tree with the smallest number, remove it from the tree, and write down the number of the vertex that was connected to it. After $n-2$ iterations there will only remain
2 vertices, and the algorithm ends.
Thus the Prüfer code for a given tree is a sequence of $n-2$ numbers, where each number is the number of the connected vertex, i.e. this number is in the interval
$[0, n-1]$.
The algorithm for computing the Prüfer code can be implemented easily with $O(n \log n)$ time complexity, simply by using a data structure to extract the minimum (for instance set or priority_queue in C++), which contains a list of all the current leafs.
After constructing the Prüfer code two vertices will remain. One of them is the highest vertex $n-1$, but nothing else can be said about the other one.
Each vertex appears in the Prüfer code exactly a fixed number of times - its degree minus one. This can be easily checked, since the degree will get smaller every time we record its label in the code, and we remove it once the degree is 1. For the two remaining vertices this fact is also true.

6.9 Prüfer's Sequence

The Prüfer sequence is a bijection between labeled trees with n vertices and sequences with $n-2$ numbers from 1 to n.
To get the sequence from the tree:

- While there are more than 2 vertices, remove the leaf with smallest label and append it's neighbour to the end of the sequence.

To get the tree from the sequence:

- The degree of each vertex is 1 more than the number of occurrences of that vertex in the sequence. Compute the degree d, then do the following: for every value x in the sequence (in order), find the vertex with smallest label y such that $d(y) = 1$ and add an edge between x and y, and also decrease their degrees by 1. At the end of this procedure, there will be two vertices left with degree 1; add an edge between them.

7 Trees

7.1 Heavy-Light Decomposition

Heavy-Light Decomposition (HLD) is a technique to decompose a tree into a set of disjoint paths. This technique is particularly useful to deal with problems which require us to do some path-queries in a tree which seemingly complicated but easy enough to be solved for a line-graph. The idea is to decompose the tree into several paths (line-graph) of disjoint vertices. Then, each path-query in the original tree might be able to be answered by queries in one or more of those paths.

An edge (a, b) is heavy if and only if size(b) size(a)/2; otherwise, it is light

8 Combinatorics

8.1 Binomial Coefficients

Binomial coefficients $\binom{n}{k}$ are the number of ways to select a set of k elements from n different elements without taking into account the order of arrangement of these elements (i.e., the number of unordered sets).
Binomial coefficients are also the coefficients in the expansion of $(a+b)^n$ (so-called binomial theorem):

$$(a+b)^n = \binom{n}{0}a^n + \binom{n}{1}a^{n-1}b + \binom{n}{2}a^{n-2}b^2 + \cdots + \binom{n}{k}a^{n-k}b^k + \cdots + \binom{n}{n}b^n$$

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

Recurrence formula** (which is associated with the famous ”Pascal’s Triangle”):

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

8.1.1 Odd numbers in the i-th line

O número de elementos ímpares na n-śeima linha do triangulo de pascal é 2^c, onde c é o número de bits na representação binária de n.

8.1.2 Properties

Binomial coefficients have many different properties. Here are the simplest of them:

- Symmetry rule:

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

- Factoring in:

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

- Sum over k:

$$\sum_{k=0}^{n}\binom{n}{k} = 2^n$$

- Sum over n:

$$\sum_{m=0}^{n}\binom{m}{k} = \binom{n+1}{k+1}$$

- Sum over n and k:

$$\sum_{k=0}^{m}\binom{n+k}{k} = \binom{n+m+1}{m}$$

- Sum of the squares:

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

- Weighted sum:

$$1\binom{n}{1} + 2\binom{n}{2} + \cdots + n\binom{n}{n} = n2^{n-1}$$

- Connection with the [Fibonacci numbers](../algebra/fibonacci-numbers.md):

$$\binom{n}{0} + \binom{n-1}{1} + \cdots + \binom{n-k}{k} + \cdots + \binom{0}{n} = F_{n+1}$$

8.2 4 fundamental problems of distribution

8.2.1 N equal balls in K equal boxes

8.2.2 N equal balls in K distinct boxes

Assuming that there can be some boxes empty:

$$\binom{N+K-1}{N}$$

8.2.3 N distinct balls in K equal boxes

8.2.4 N distinct balls in K distinct boxes

9 Bitwise

9.1 Binary to gray code

$G_n = B_n$
$G_{n-1} = B_n \oplus B_{n-1}$
$G_i = B_i \oplus B_{i-1}$
...
$G_1 = B_2 \oplus B_1$

9.2 Gray code to binary

$B_n = G_n$
$B_{n-1} = B_n \oplus G_{n-1} = G_n \oplus G_{n-1}$
...
$B_1 = B_2 \oplus G_1 = G_n \oplus G_1$

10 Game Theory

10.1 Impartial Games

To be considered a impartial game following rules must be true:

1. The available moves win/loose depends only on the state of the game, in other words, the only difference between the two players is that one of them moves first

2. Additionally, we assume that the game has perfect information, i.e. no information is hidden from the players (they know the rules and the possible moves).

3. It is assumed that the game is finite, i.e. after a certain number of moves, one of the players will end up in a losing position — from which they can't move to another position. On the other side, the player who set up this position for the opponent wins. Understandably, there are no draws in this game.

Such games can be completely described by a directed acyclic graph: the vertices are game states and the edges are transitions (moves). A vertex without outgoing edges is a losing vertex (a player who must make a move from this vertex loses).
Since there are no draws, we can classify all game states as either winning or losing. Winning states are those from which there is a move that causes inevitable defeat of the other player, even with their best response. Losing states are those from which all moves lead to winning states for the other player. Summarizing, a state is winning if there is at least one transition to a losing state and is losing if there isn't at least one transition to a losing state.
Our task is to classify the states of a given game.

10.2 Sprague-Grundy Theorem

The Sprague-Grundy Theorem states that every impartial game is equivalent to a pile of a certain size in Nim. In other words, every impartial game can be solved as Nim by finding their corresponding game.
Basically, for a game situation A and its SG function value $g(A)$:

1. $g(A) = 0$ if and only if A is a must-lose situation. Otherwise, $g(A) \in \mathbb{Z}*$

2. If A can be divided into n sub-situations $x_1, x_2, ..., x_n$, then $g(A) = g(x_1) \oplus (x_2) \oplus ... \oplus g(x_n)$

3. If A can be converted to situation B_1 or B_2 or ... or B_n by only one operation, then $g(A) = mex(g(B_1), g(B_2), ..., g(B_n))$ where function $mex(S)$ is defined as the smallest non-negative integer that does not appear in S. For example, $mex(0,1,2,4) = 3, mex = 0, mex(0,1,2,4) = 3, mex() = 0$

10.3 Nim variation Subtract game

Work just like nim but instead remove any number of objects you can remove at most K, this game can be seen as a nim game but before computing the num-sum you need to take the size of each pile module $K+1$, the optimal way to play it is by taking K at each turn.

11 Algebra

11.1 Negative numeric bases

Negative numbers in negative bases have an even number of digits, while positive numbers in negative bases have an odd number of digits.
Can be found using the common algorithm but need to be careful about the modulus operators in most languages.
iagorrrrr

12 Others

12.1 Critérios de divisibilidade

12.1.1 7

Para verificar a divisibilidade de um número por 7, siga a seguinte regra:

1. Pegue o número em questão.
2. Remova o último dígito (unidade) do número.
3. Dobre o valor removido no passo anterior.
4. Subtraia o valor dobrado do número restante.
5. Se o resultado da subtração for divisível por 7, o número original é divisível por 7.

Exemplo:
Suponha que desejamos verificar a divisibilidade do número 413 por 7.

1. Remova o último dígito (3) e dobre-o, obtendo 6.
2. Subtraia 6 do número restante (41 - 6 = 35).

12.1.2 11

$$n \text{ é divisível por 11} \iff \sum_{i=1}^{k} a_{2i-1} - \sum_{i=1}^{j} a_{2i} \text{ é divisível por 11}$$

onde a_i é o i-ésimo dígito do número n, k é a quantidade de dígitos ímpares, j é a quantidade de dígitos pares.
Exemplo:
Suponha que desejamos verificar a divisibilidade do número $n = 7923$ por 11.

$$k = 2, \quad j = 2$$

$$\text{Soma dos dígitos ímpares: } 7 + 3 = 10$$

$$\text{Soma dos dígitos pares: } 9 + 2 = 11$$

$$\text{Subtração: } 10 - 11 = -1$$

Como -1 não é divisível por 11, o número 7923 não é divisível por 11.

12.1.3 13

$$13|x \equiv 13|4 \cdot (x\%10) + \lfloor x/10 \rfloor \tag{11}$$

Em outras palavras 13 divide x se o quádruplo do último algarismo somado com o número sem este algarismo for divisível por 13.

12.1.4 17

$$17|x \equiv 17|\lfloor x/10 \rfloor - 5 \cdot (x\%10) \tag{12}$$

Em outras palavras 17 divide x se o a diferença entre o quíntuplo do último algarismo e o número sem este algarísmo for divisível por 17.

12.1.5 19

$$19|x \equiv 19|\lfloor x/10 \rfloor + 2 \cdot (x \mod 10) \tag{13}$$

Em outras palavras 19 divide x se o dobro do último algarismo de x somado a o número restante de x é divisível por 19.

12.1.6 23

$$23|x \equiv 23|x/10 + 7 \cdot (x \mod 10) \tag{14}$$