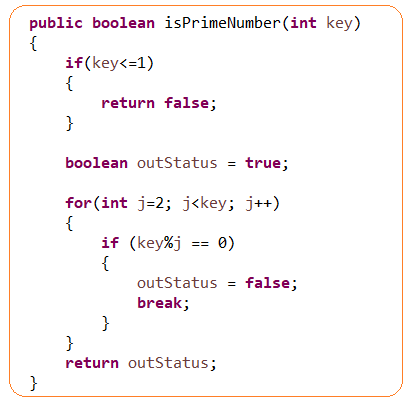
# **MATH**

# **PRIME NUMBER**

**Prime numbe**r is a number that has two dividers - one and itself. Prime numbers starts from 2

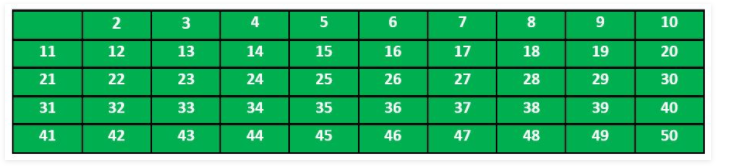
Example: 2, 3, 5, 7, 11



# **SIEVE OF ERATOSTHENES**

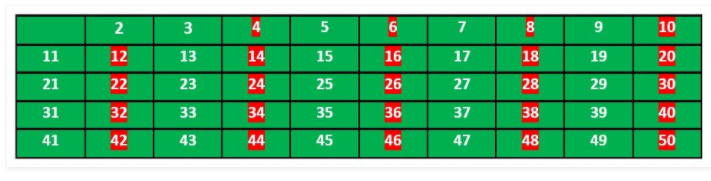
SIEVE OF ERATOSTHENES – the ancient method of finding prime numbers

1. We generate a range of numbers what we want to checks



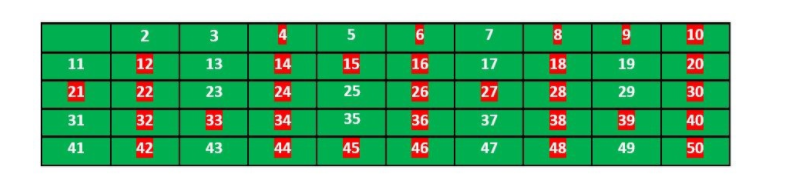
1. Divide all numbers by 2. Those which are not dividable (n%2 != 0) will leave as not touchable

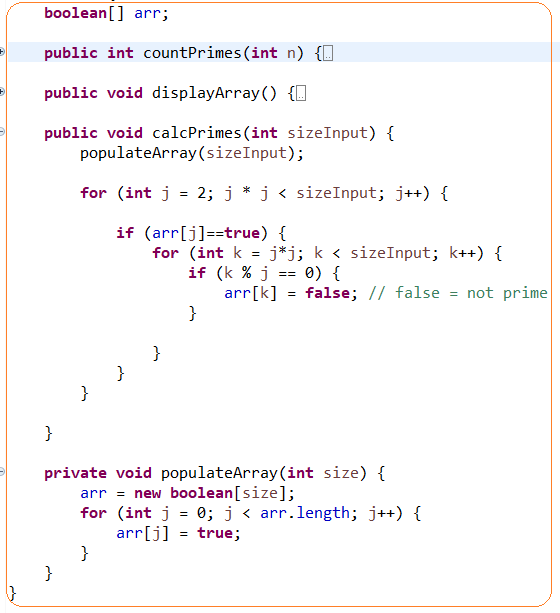
2\*2=4, 2\*3=6, 2=8, 2\*5=10, 2\*6=12



1. Repeat the same for 3

3\*3=6, 3\*4=12, 3\*5=15, 3\*6=18

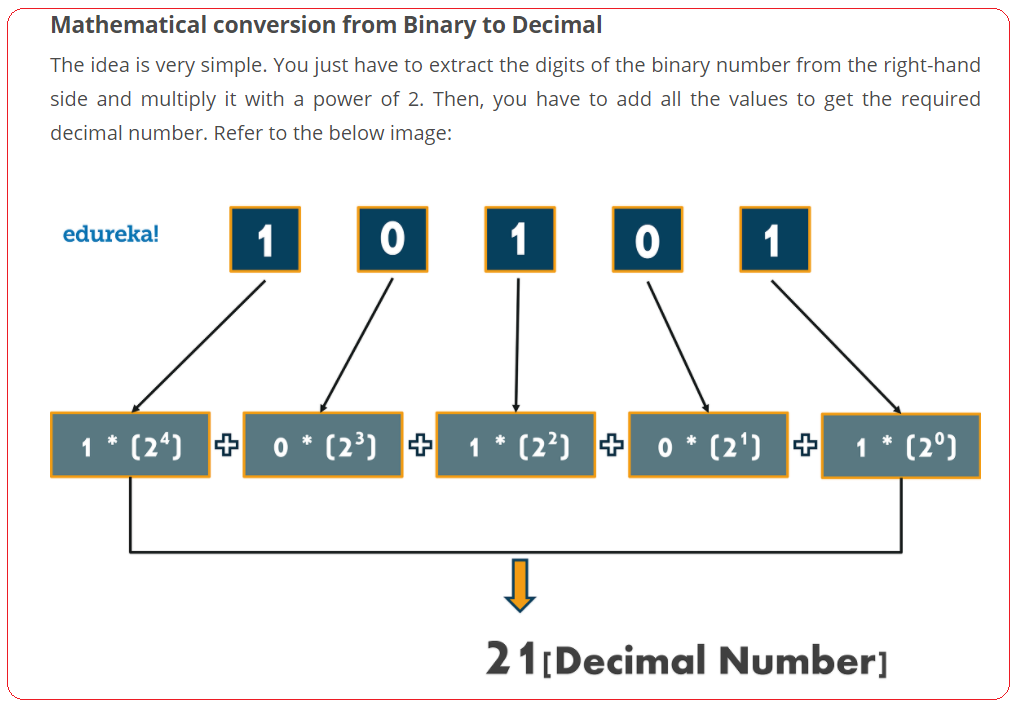


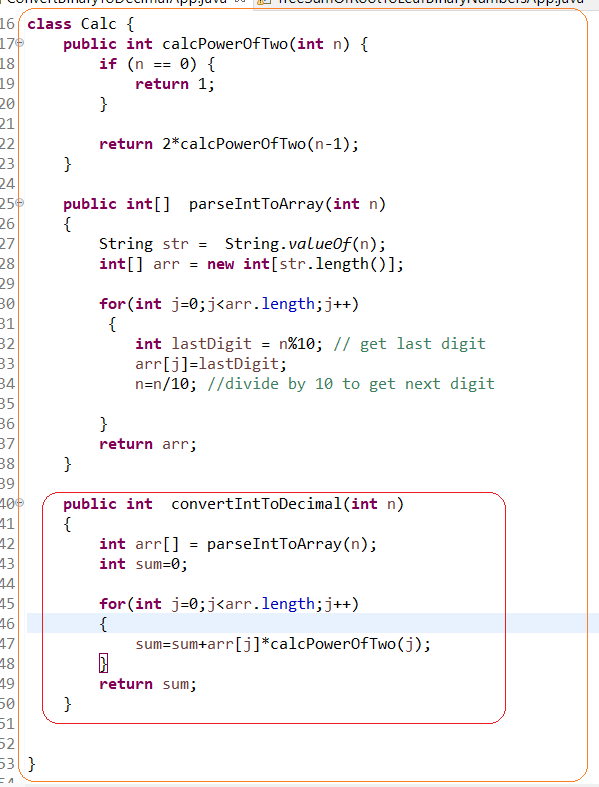


# **CONVERT\_BINARY\_TO\_DECIMAL**

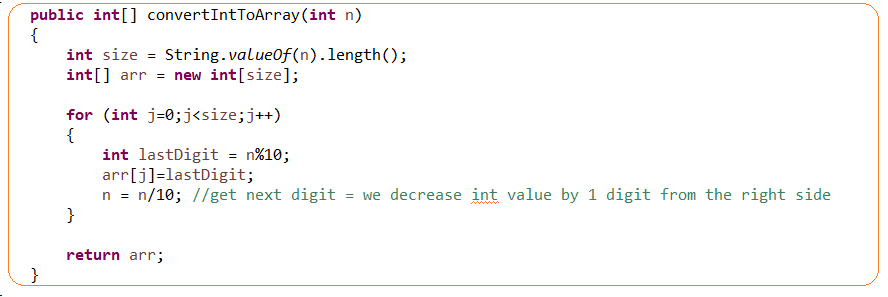
**Binary number**: A binary number is a number expressed in the base-2 numeral system or binary numeral system. This system uses only two symbols: typically 1 (one) and 0 (zero).

**Decimal number:** The decimal numeral system is the standard system for denoting integer and non-integer numbers. It is also called base-ten positional numeral system.

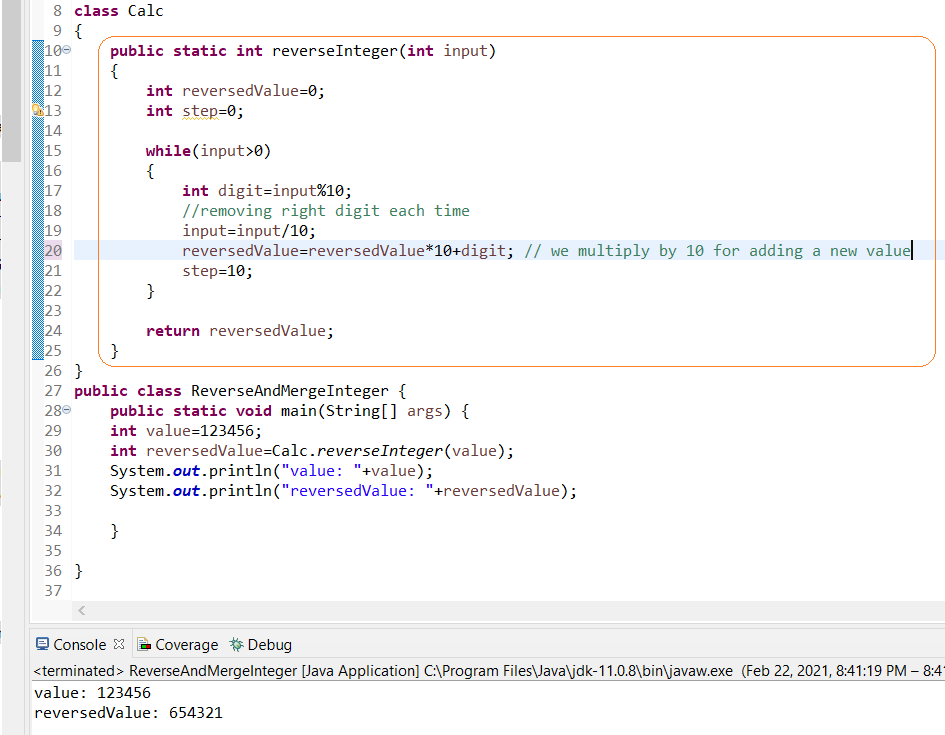




# **CONVERT INTEGER TO ARRAY OF INTEGERS**



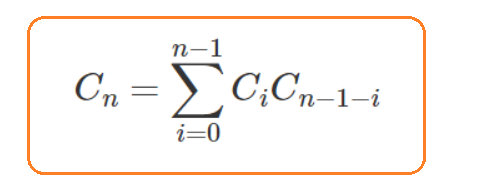
## **REVERSE\_NUMBER\_AND\_MERGE\_DIGIHT**

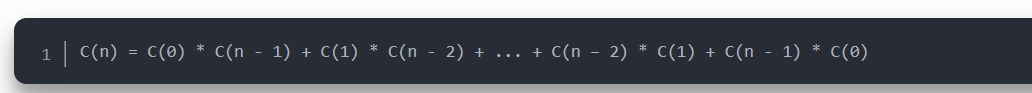


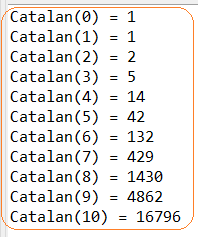
## **CATALAN NUMBER**

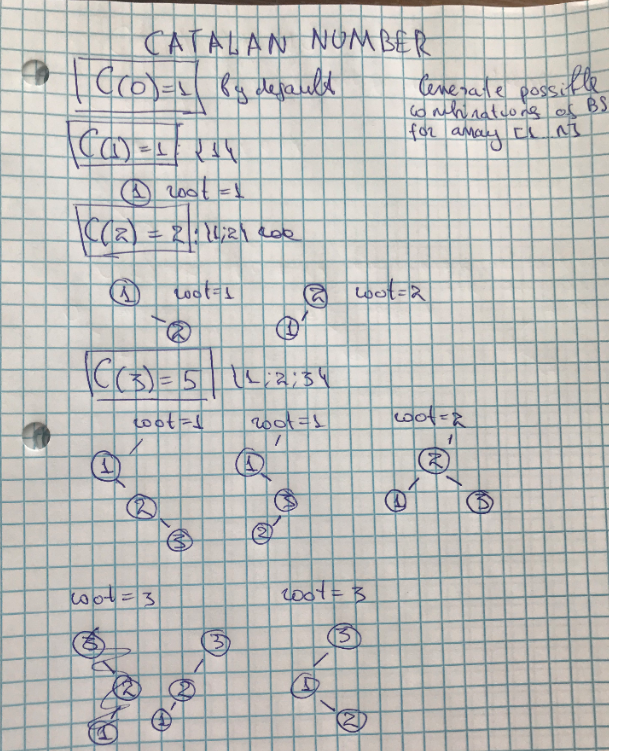
<https://coldfunction.com/mgen/p/3r>

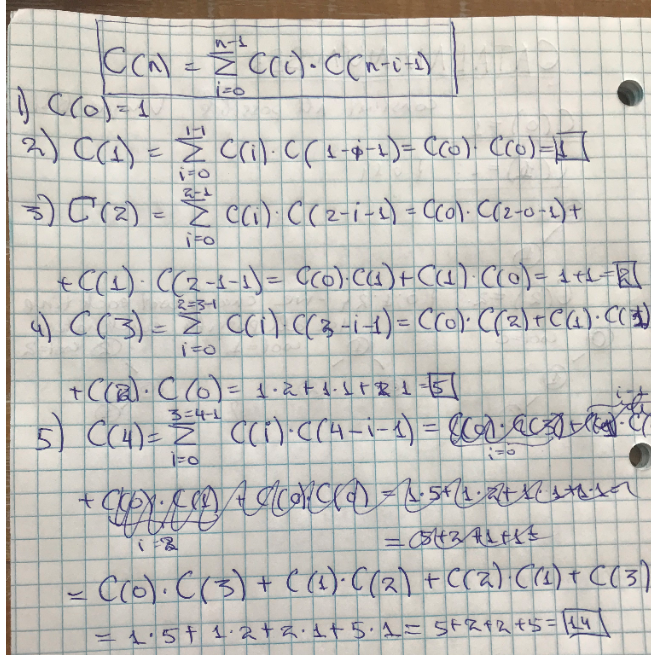
**Catalan number** is a sequence of numbers that can be calculated by formula. It is used in many problems. One of them is count all possible unique binary trees

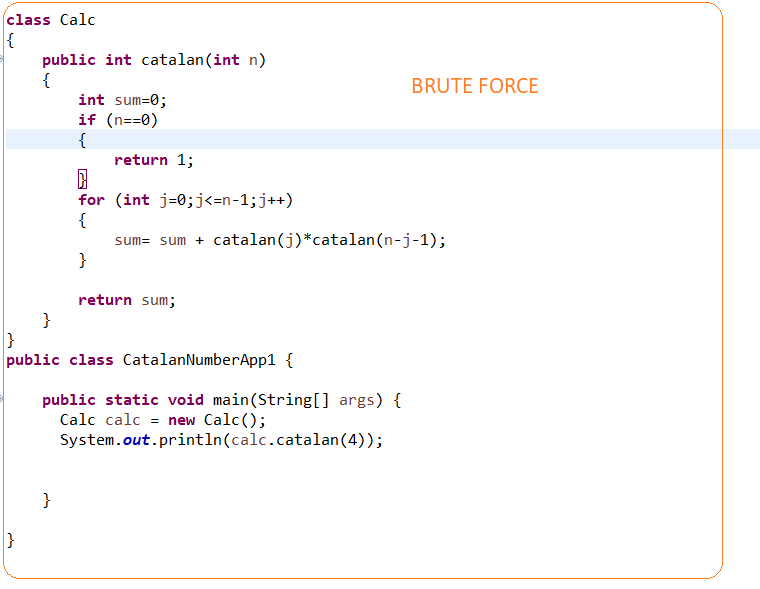


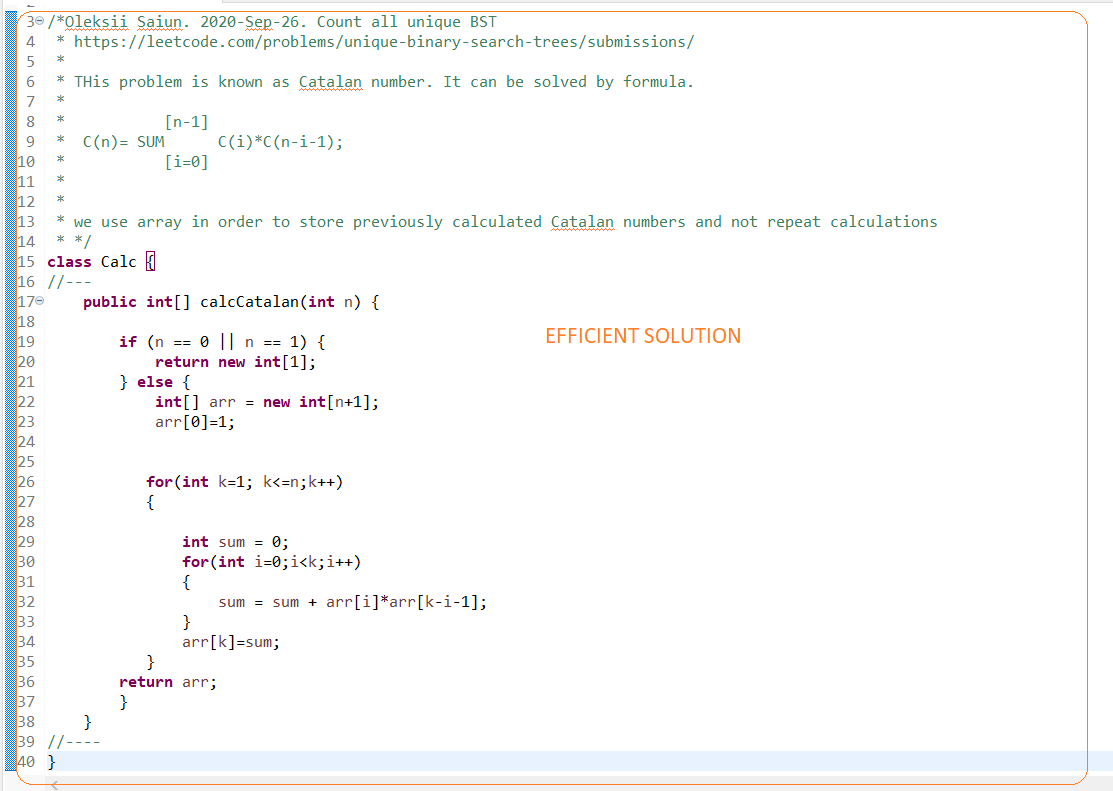


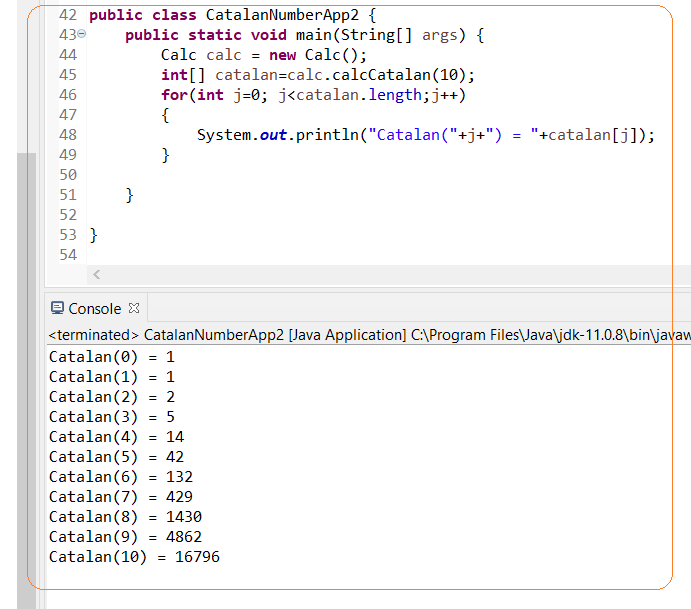




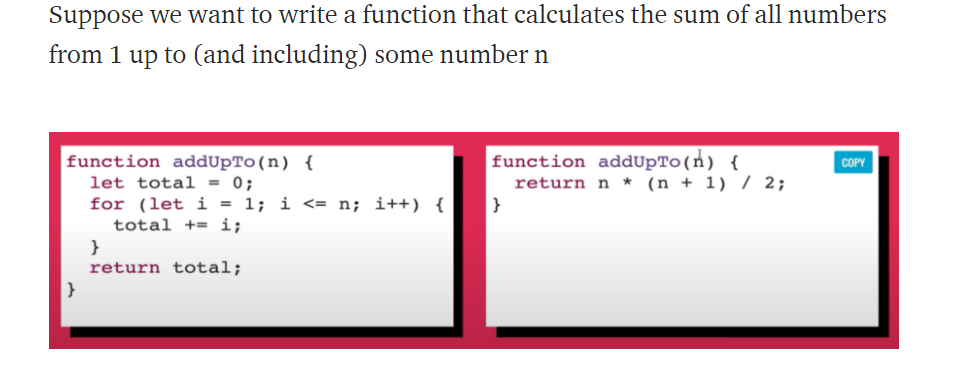








## **SUM OF ALL NUMBERS 1..N**



## **LOGARITHM**

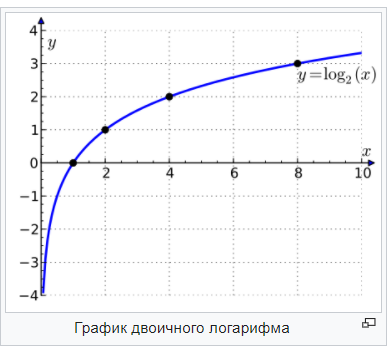
Logarithm of number [Y] base [B] is equal to [X]

**BY = X ⬄ logBY=X**

**23 = 8 ⬄ log28=3**

**Logarithm** (BY = X ⬄ logBY=X) is exponent that we need to raise base B to power Y to get number X.

In computer science log2Y = logY – base 2 disregards and everything is considered as logarithm base 2 by default.



**2y = x ⬄ log2y=x**

|  |  |
| --- | --- |
| Y | X |
| 0 | 1 |
| 1 | 2 |
| 2 | 4 |
| 3 | 8 |
| 4 | 16 |

## **LOGARITHM IN COMPLEXITIES**

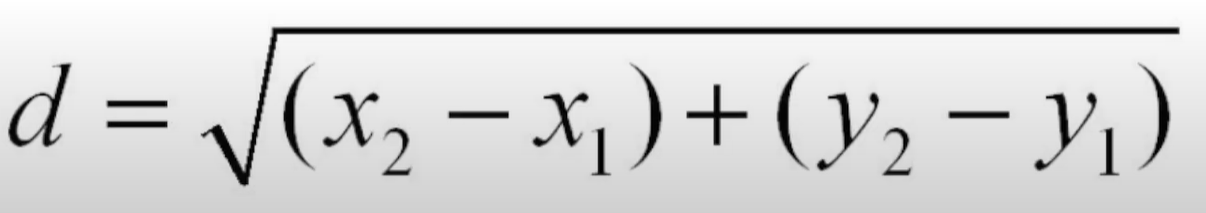
When there is a problem connected with a binary tree or a binary search it is usually connected with O(log(n)) (in a variations)

How many times we can split this array(of length 8)?

## 

## **FIND DISTANCE BETWEEN TWO POINTS A(X1,X2) AND B(Y1,Y2)**

A **distance (Euclidean distance)** between two coordinates in 2D can be calculated by formula:



**Example**:

The problem “*K Closest Points to Origin*”

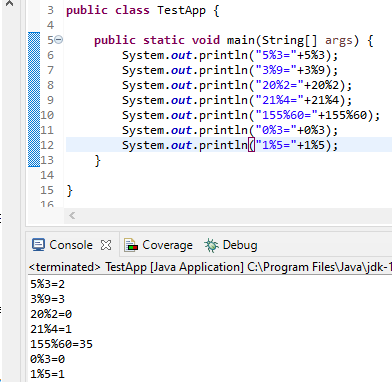
<https://leetcode.com/problems/k-closest-points-to-origin/submissions/>

Is solved using this formula

<https://github.com/oleksiisaiun/JAVA/tree/master/ALGORITHM/DATA-STRUCTURES/HEAPS/K_CLOSEST_POINTS_TO_ORIGIN>

## **MODULE**

**Modulo Operator (% sign)** - returns the remainder of the two numbers after division



Modulo Operator properties

**(x+y)%60=0 <=> ((x)%60 + (y)%60)%60=0**

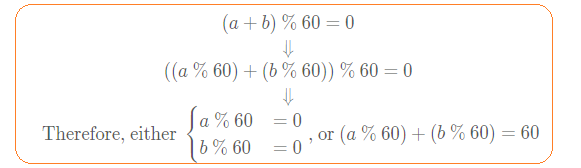
**<=> it equals two cases:**

**x%60=0**

**y%60=0**

**or <=> (x)%60 = 60 - (y)%60)**

<https://leetcode.com/problems/pairs-of-songs-with-total-durations-divisible-by-60/solution/>



## **FORMULAS**

