1) Divide and conquer
2) Greedy
3) Dynamic Programming.

} Problems on these paradigms.

All studied algorithm takes $O(n^c)$ time. for some constants.

These problems have efficient algorithms.

Polynomial time.

Question: What happens when one cannot find an efficient algorithm for a problem?

→ Showing a problem has an efficient algorithm is relatively easier.

- one needs to design one algorithm.

→ Proving that no efficient algorithm exists for a particular problem is difficult.

Question: How can we prove the non-existance of something?

# Two categories

## Solvable

Some algorithm exists.

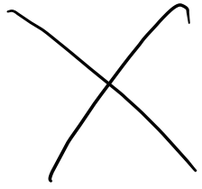
✓

$O(n^c)$

$O(2^n)$

## Not solvable

No algorithm possible.



Halting problem.

# NP- complete problem

- A thousands of problems for which
- It is not known whether the problems have efficient solutions or not.
- It is known that if any one of the NP- complete problems has an efficient solution that all of them have efficient solution
- There is a large number of tools exist to prove a new problem to be NP- complete.
- The problem of finding an efficient solution to an NP- complete problem is known as $P \neq NP$?

Million dollar question.

claymath.org

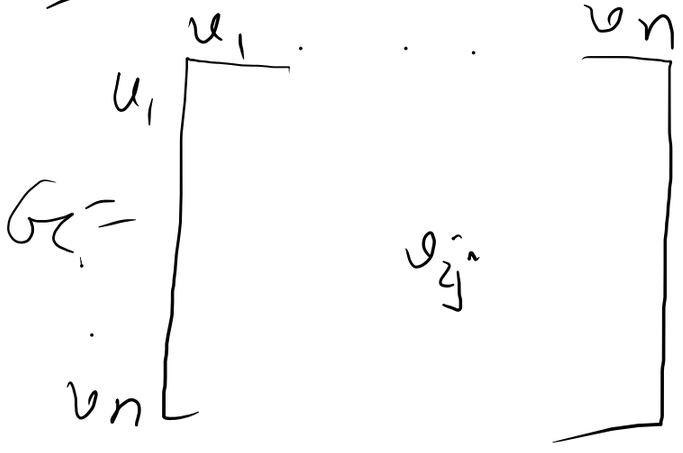
# Input to a problem

$n$ be the length/size of the input.

$f(n)$ its time complexity.

# Encoding of the input:

Ex^m How a graph is encoded ? Adjacency matrix.




$a_{ij} \leftarrow$ binary string of $v_{ij}$

$G_c$ can be encoded as.

$a_{11}\ a_{12} \ldots a_{1n}\ a_{21}\ a_{22} \ldots \ldots a_{nn}$

length of the string $n^2 \cdot b$ where $b = \max_{ij} |a_{ij}|$

In general: the input of any problem can be encoded as a binary string.

Input size: minimum number of bits $\{0,1\}$ needed to encode the input of the problem.

Exm Sorting problem

what is the input size.?

Problem a sequence of numbers $a_1, a_2 \ldots, a_n$

Rearrange to make these nondecreasing.

Input size: $b_i \leftarrow$ binary encoding of $a_i$:

$$k = \max_i |b_i|$$

Input size: $k \cdot n$

# Integer multiplication

$a, b$ $\quad a \times b$

Input size $\leq 2 \cdot n$ $\quad n = \max\{b_1, b_2\}$

## Decision problems

**Defn** problems that have yes or no answer.

## Optimization problems:

certain configurations need to be optimize

minimize maximize

Ex^m MST

| optimization | Decision |
| --- | --- |
| • Given an edge weighted graph G | • G and an a number t |
| • Find a tree T that spans all the vertices of G | • - - - - |
| • Weight of T is minimum | • Decide whether the weight of the tree is at most t or not |

H.W Find the maximum of 10 numbers.