NP vs P

classes of computational problems.

(polynomial time)

P = "class of problems for which one can find the solution in polynomial time",

$O(n^c)$ for some c

efficiently

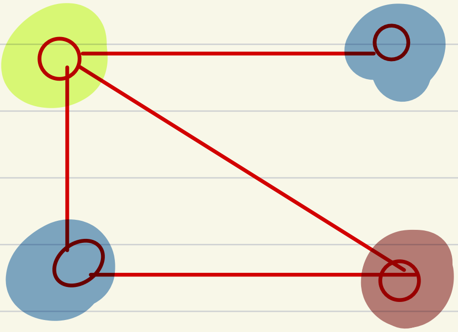
NP = "class of problems for which one can verify a solution in polynomial"

3-COLORING $\in NP$

INPUT: Graph $G=(V,E)$

SOLUTION: A coloring of V with 3colors $\{R,B,G\}$ such that every edge recieves two different colors

#3COL:
INPUT: Graph $G=(V,E)$
SOLUTION: Number of 3-colorings



Claim: 3COLORING $\in NP$
(verify a solution in polynomial time)

PROOF:

```
VERIFY( INPUT $G=(V,E)$ , SOLUTION $c: V \to \{R,B,G\}$ )
    for each edge $(u,v) \in E$
        check that $c(u) \neq c(v)$
```

FACTORIZATION $\in$ NP

INPUT: An n bit number N.

SOLUTION: Two numbers $p, q > 1$ so that $p \cdot q = N$

Claim:
FACTORIZATION $\in$ NP

VERIFY (INPUT N , SOLUTION $p, q,$)

1) Check $p, q > 1$ ←

2) Check $N = p \cdot q$ ← $O(n^2)$ time

$P \subseteq NP$

(find the solution) → verify it

MINIMUM SPANNING TREE $\in P$

INPUT: Graph $G = (V, E)$ with weights w_e

Solution: Minimum Spanning Tree T

MST $\in$ NP

Proof:

$$\text{Verify}\Big(\underset{\text{weights}}{\overset{\text{Input}}{\text{Graph } G=(V,E)}},\ \overset{\text{Solution}}{T}\Big)$$

{

MST $T^* \leftarrow$ Kruskals Algorithm(G)

Check $\text{cost}(T) = \text{cost}(T^*)$

}

RUDRATA CYCLE

(HAMILTON CYCLE)

INPUT: A Graph G = (V,E)

SOLUTION: A cycle visiting every node exactly once.

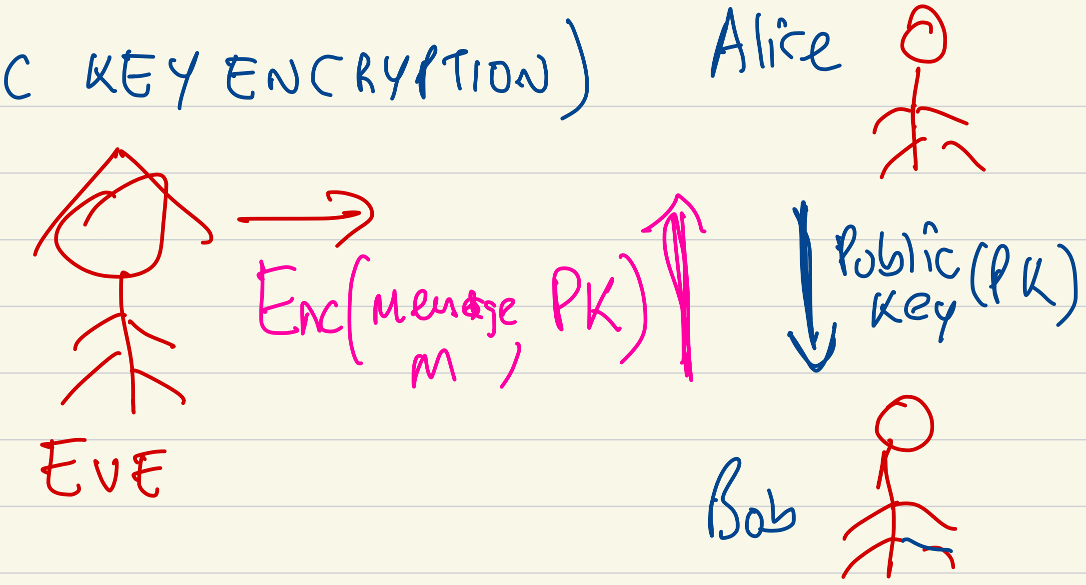
RUDRATA CYCLE $\in NP$

Proof:

```
VERIFY ( INPUT                 Solution
         Graph G = (V,E)   ,   A cycle C )
{


}
```

BREAKING RSA (PUBLIC KEY ENCRYPTION)



INPUT: Public key PK

Ciphertext: ENC(PK, m)

SOLUTION: Message m

BREAKING RSA ∈ NP:

Proof: $$\text{VERIFY}\left(\overset{\text{INPUT:}}{PK,\ \text{ciphertext}}\ ,\ \overset{\text{Solution}}{\text{Message } m}\right)$$

{

Check Ciphertext $=$ ENC(m, PK)

}

PROBLEMS NOT IN NP

- → Counting Problems
- → Games
- → Halting Problem
- →

MINIMUM TRAVELLING SALESMAN PROBLEM: (MIN TSP)

INPUT: A Graph $G = (V, E)$ with edge weights

SOLUTION: Smallest weight cycle that visits every node exactly once.

MIN TSP $\overset{??}{\notin}$ NP

BUDGET TSP: $\in$ NP

INPUT: 1) A Graph $G = (V, E)$ with edge weights

2) BUDGET B.

SOLUTION: Cycle that visits every node exactly once, with cost $\leq B$

(OPTIMIZATION) (NP-SEARCH PROBLEM)

$\underline{\underline{\text{MIN TSP}}} \notin NP \xLeftrightarrow[\text{alg}]{\text{efficient}} \underline{\underline{\text{BUDGET TSP}}} \in NP \Longrightarrow NP$

(P^{NP})

$\Downarrow$

(DECISION VARIANT)

DECISION BUDGET TSP:

INPUT: 1) A Graph $G = (V,E)$ with edge weights

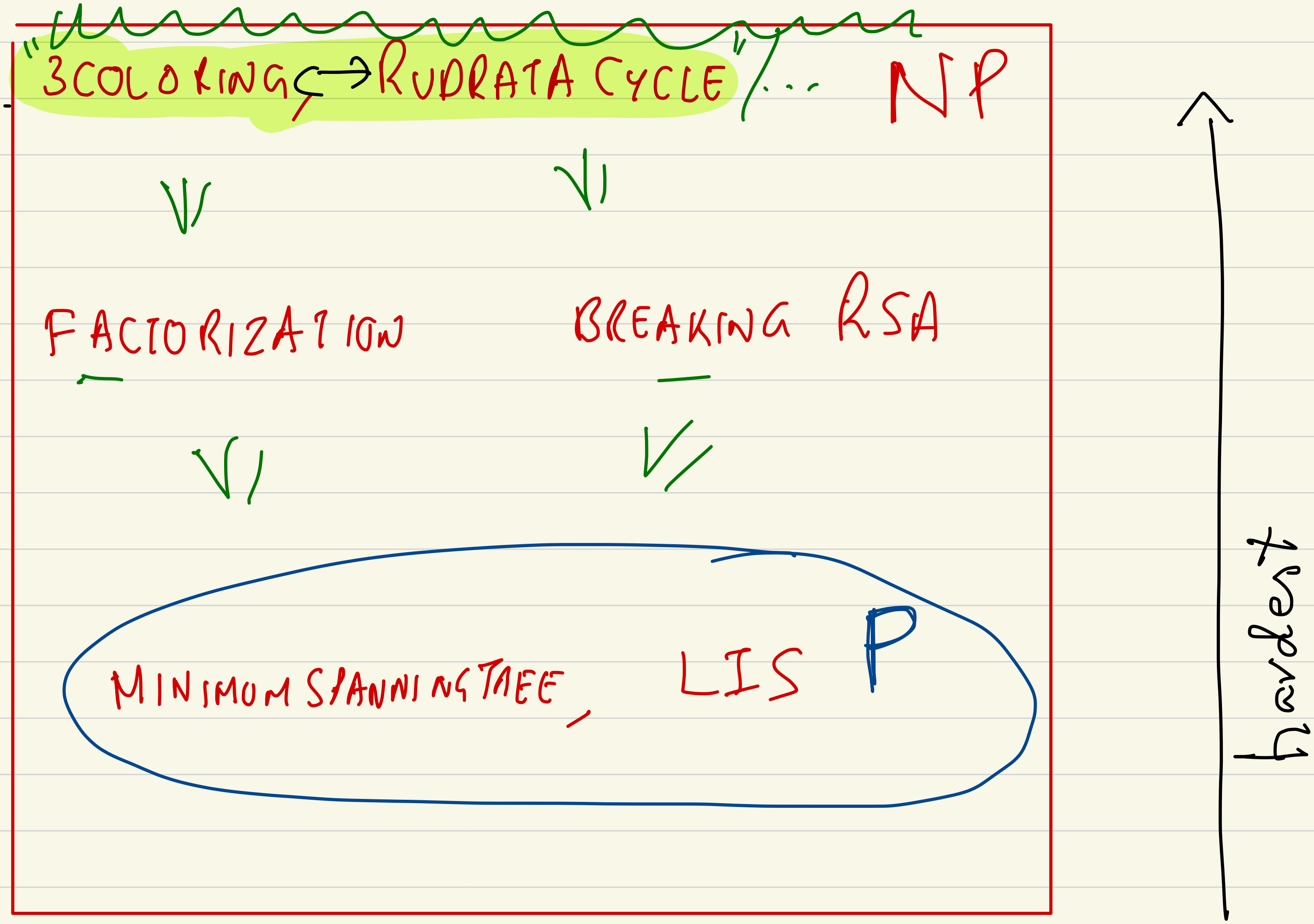
2) BUDGET B.

SOLUTION: Does $\exists$ Cycle that visits every node exactly once, with cost $\leq B$??

"NP-complete Problems"

=

"hardest problems within NP"



$NP \neq P$??

REDUCTIONS: technique to compare difficulty of problems.

FACTORIZATION $\leq_{\textcircled{P}}$ 3-COLORING ??

time complexity

PROBLEM A $\leq_P$ Problem B

“Problem A reduces in polynomial time to Problem B”

<u>**IF**</u>

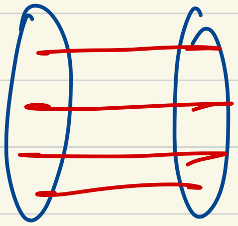
“Use any algorithm for B to efficiently solve A”

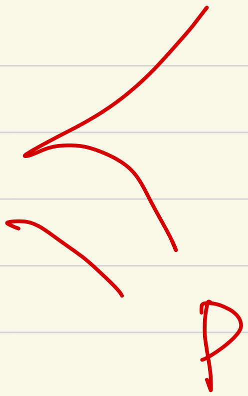
⇒ “Problem A is ~~easier~~ at most as hard than Problem B”

MATCHING

INPUT: Bipartite Graph G

SOLUTION:

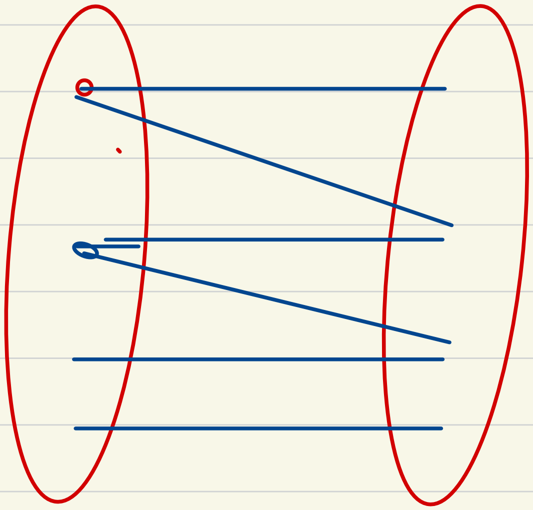




MAXIMUM FLOW

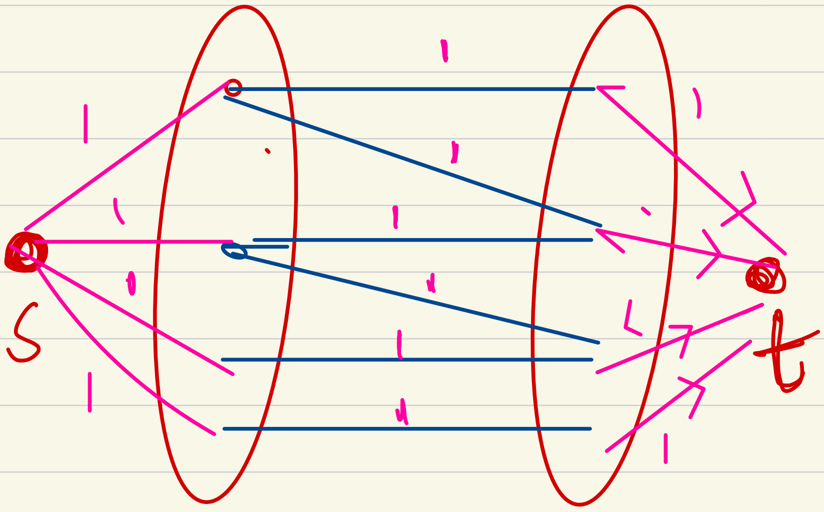
INPUT: 1) Graph with capacities s, t
2) Total flow = B

SOLUTION: A flow of value B.



$G=(V,E)$

$\Longrightarrow$



$G'=(V,E)$

Recover a matching.

$\Longleftarrow$

1) Compute Maximum Flow in G' from s to t.

RUDRATA CYCLE

INPUT: Graph $G=(V,E)$

SOLUTION: A cycle through all vertices exactly once

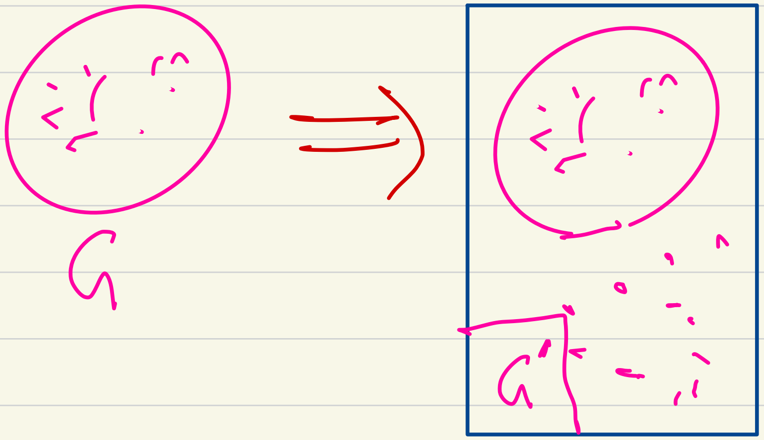
$\leq_P$

HALF-CYCLE

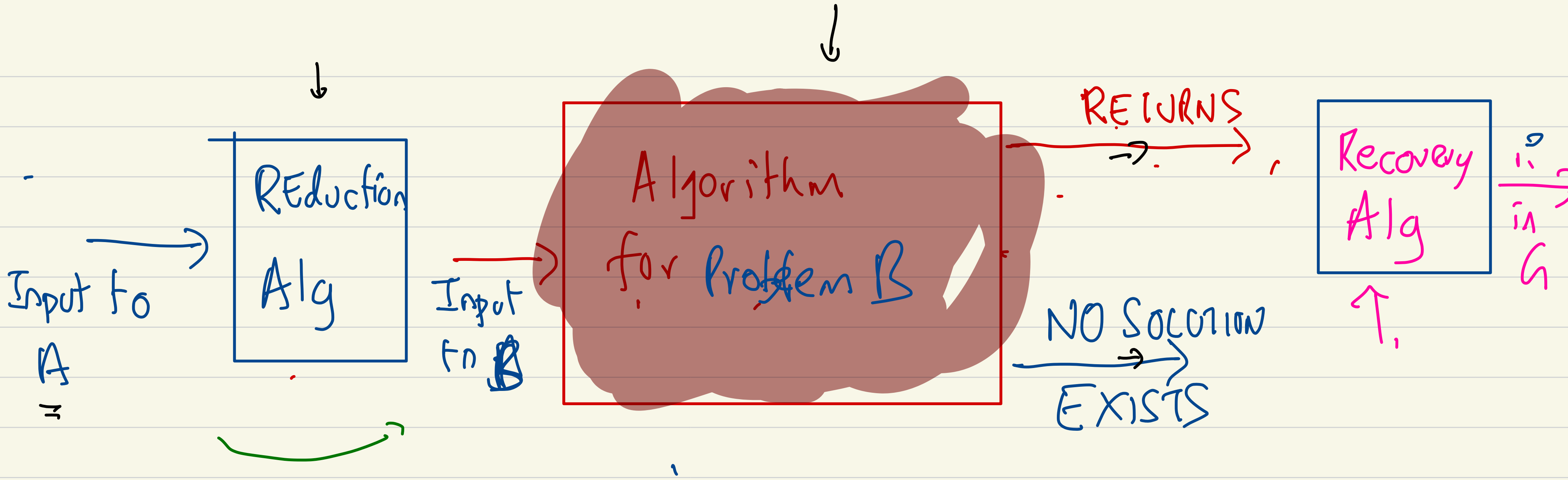
INPUT: Graph $G'=(V',E')$

SOLUTION: A cycle through $|V'|/2$ vertices.

"Any Algorithm for HalfCycle can be used solve Rudrata Cycle"



$G' = G$ with n isolated vertices



INGREDIENTS:

1) Reduction Alg: Convert instance of problem A → to problem B

2) Recovery Alg: reconstruct solution to A from a solution to B

Solution to B ⇒ Solution to A

3) No solution to B ⇒ No Solution to A

Solution to A ⇒ Solution to B.

→ RUDRATA CYCLE is NP-complete
(HAMILTON)

→ CIRCUIT SAT is NP-complete

(Mother of all NP-completeness results)

Every problem in NP $\leq_p$ Circuit Sat