Lecture-04

7/9/2024.

Satisfiability Problem : SAT

Boolean ckt — 1 output

If there is any single combination of inputs so that the equation equates to true.

— (any) arbitrary ckt

Easy : # of inputs যদি কম হয়।

n inputs → # of combinations $= 2^n$

→ But we want to perform this check in polynomial time.

Genius trick/challenge.

CNF

SAT : special case

conjunctive normal form

product of Sums.

$Y = (x_4 + \bar{x}_3 + x_4)(x_2 + x_5 + \bar{x}_3)(+ + .)$

$(+ . + .)$.

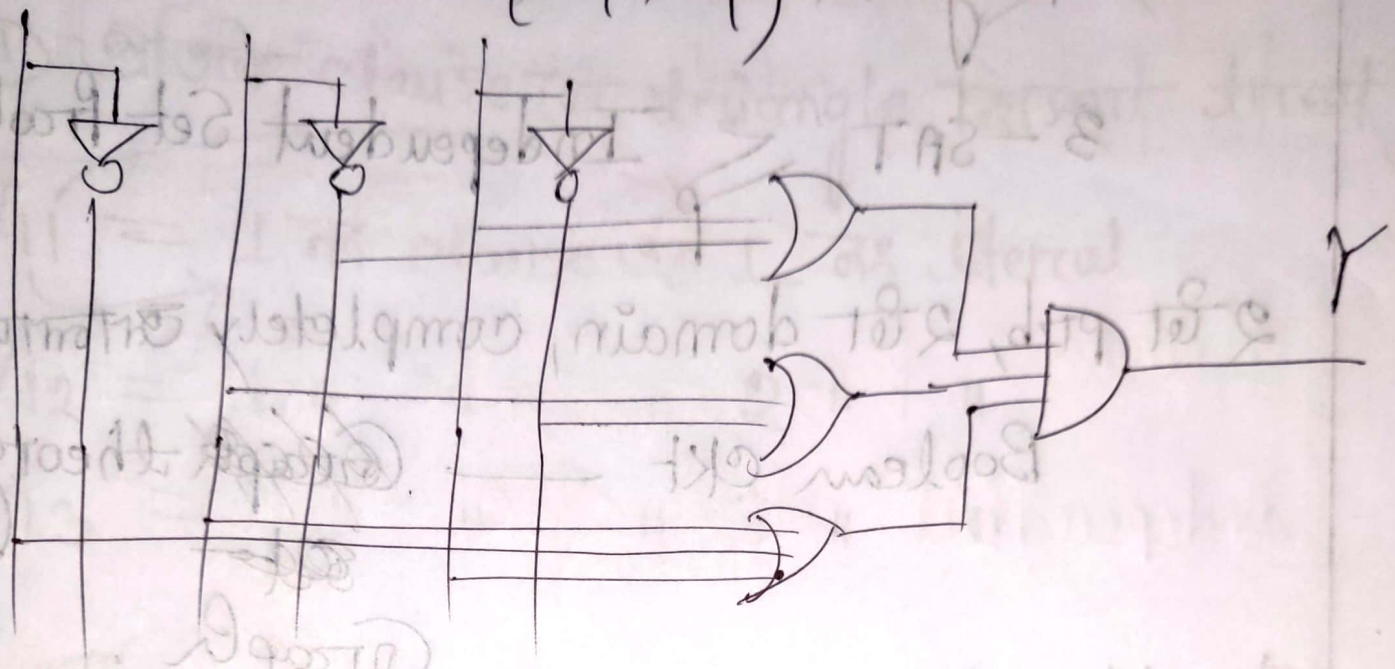
Y → output

$x_4, x_3, x_4, x_2, x_5, x_3$ → boolean variables

→ clause

variable = $\{x_i\}$

literal / term = $\{x_i, \overline{x_i}\}$



Special case of CNF SAT : 3-SAT

At most 3 टा term / literal थाकबे in each clause.

— No limitation at the # of Variables.

II 3-SAT is reducible to Independent Set Prob in polynomial time.

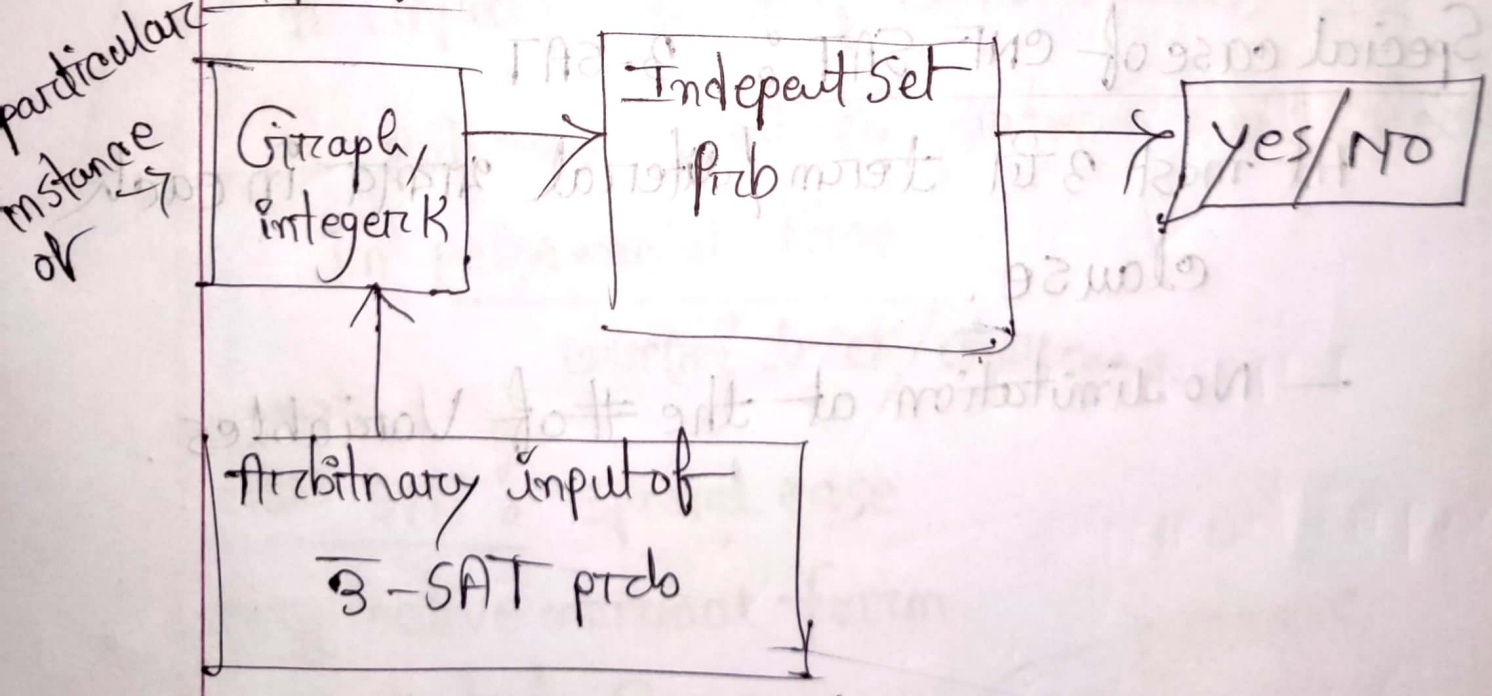
$$3\text{-SAT} \leq_p \text{Independent Set Problem}$$

2 ते prob, 2 ते domain, completely ~~अलगत~~ ।

Boolean CKt —— ~~Graph~~ ~~Set~~ theory.

Graph.

Step-1:



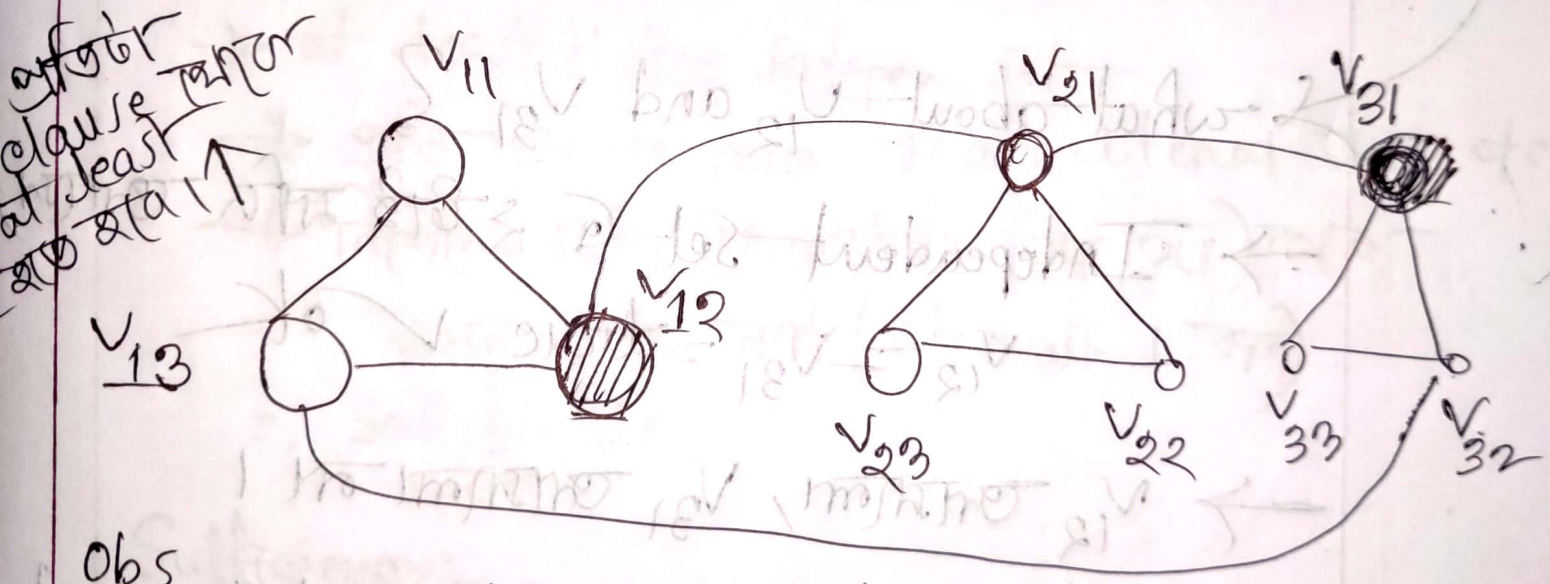
$Y = (x_1 + \bar{x}_1 + x_4)(x_2 + x_3 + \bar{x}_5)(\bar{x}_2 + \bar{x}_4 + x_5)$

① प्रत्येक clause के triangle शिखर treat करना।

V_{11} = 1 नं clause का 1 नं literal

V_{12} = 1 " " " 2 " "

V_{13} = 1 " " " 3 " "



Obs

① 3-SAT : x_i, $\bar{x}_i$ both can't be 1.
⇒ Ind set में दो vertex कभी भी ऐसा नहीं आना चाहिए if ∈ some edge's endpoints.

② प्रत्येक triangle से 1 ही vertex आना चाहिए।

# of clause यदि k हो, then # number of vertices in ind. set = k.

Now, in the constructed graph,

# of Ind. set $\leq$ # of clause (K)

$\square$ solves whether # of Ind. set $\geq K$

Exactly K তে গিয়ে ঠেকবে।

Lemma: Y is satisfiable iff G has an independent set of size at least K.

→ what about V_{12} and V_{31}?

→ Independent set এ হাই যদি আসে,

$V_{12} = V_{31} =$ true ✓ ok

→ V_{12} আসলো, V_{31} আসলো না।

V_3 = true / No Assignment imposed on V_3.

Not like আগে set এ আসার আগে true, অথবা false.

∴ হই same variable literal এ কোনো constraints impose করতে হচ্ছে না।

x_i, $\bar{x}_i$ ২টাই set এ থাকতে পারবে না।

∴ তাদের মধ্যে just edge দেয়া হচ্ছে।

same ২টা literal এ (v_{12}, v_{31}) কোনো constraints impose করার দরকার নাই।

Necessity:

~~if G has ... at least k~~

→ at least K size independent set

→ প্রতি clause থেকে ১টা করে literal আসে etc.

→ নির্বাচিত নোড গুলোকে সংশ্লিষ্ট $(x_i, \bar{x}_i)$ কার

এমন ২ জন নাই (at most 1 জন).

∴ Satisfiable.

Sufficiency:

input conversion time : polynomial prob ✓

output " " : polynomial prob ✓

Lecture-05

8/9/2024

Class P and NP: prbs are classified based on hardness.

let, Π is a prob

A algo to solve Π : $O(n^k)$ ← constant

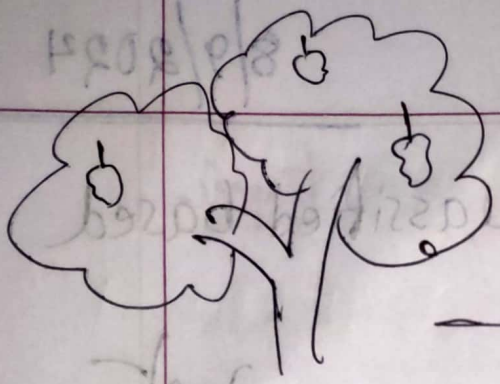
input size

- then, A is an efficient algo.
  - k can be very large, but must be constant

then it is poly. time complexity

- but in practice, k এর large হলে practical না।

- Machine will be deterministic.



পাকা আম আছে কি না
— আমায় constant time এ বলে দেয়া
Non-deterministic

$\frac{n}{n}$ tree traversal in linear time
— Deterministic.

Tractable	Intractable
যেগুলোর জন্য poly time algo আছে।	নাই & unlikely যে থাকবে।

P set of all decision pbs solvable by determi- algo in poly time.

(rigorous search)

NP non-deterministic
. . . . . poly time

(just আমায় বলে দেয়া)

যেটা deter. m/c দিয়ে poly দিয়ে solvable,
ofc সেটা non-det., " " " "

Non-deterministic Algo

Guessing : sol^n guess করা হয় । — poly.

Verification : sol^n correct কি না । — poly.

Cook's theorem :

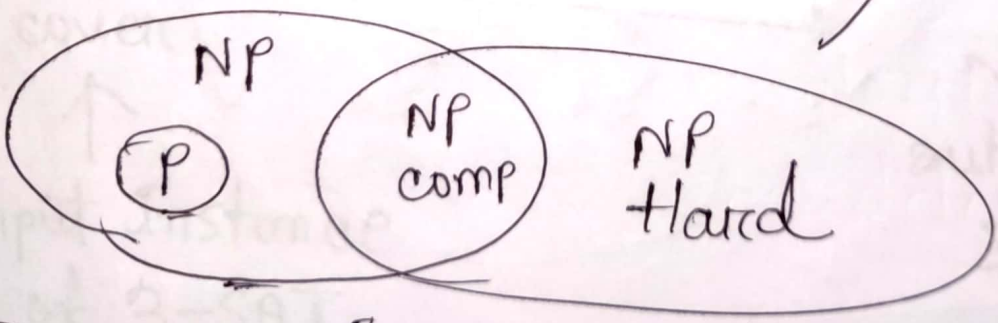
satisfiability prob is in P if & only if P=NP ?

NP-Hard :

SAT $\leq_p$ X , then X is NP hard.

NP-Complete :

X is NP-hard
and $X \in NP$



To prove X is NP-complete :

(1) $X \in NP$ → Non-deterministic m/c তিনে poly. time এ solve করা যায় ।

(2) X is NP hard

But m/c গুলো are deterministic এগুলো just try করে তো poly. time এ verifiable ?

II Vertex Cover is in NP. ↓

Verifiable কি না in poly time?

vertex গুলা delete করা হলে ∈ Vertex cover last এ আর কোনা edge থাকবে না।

∴ যে কোনো algo/method লিখতে হবে যেটা Verify করা যায় in poly time.

Lecture – 06 10/9/24

Vertex Cover prob is NP-Complete.

prob. is in NP

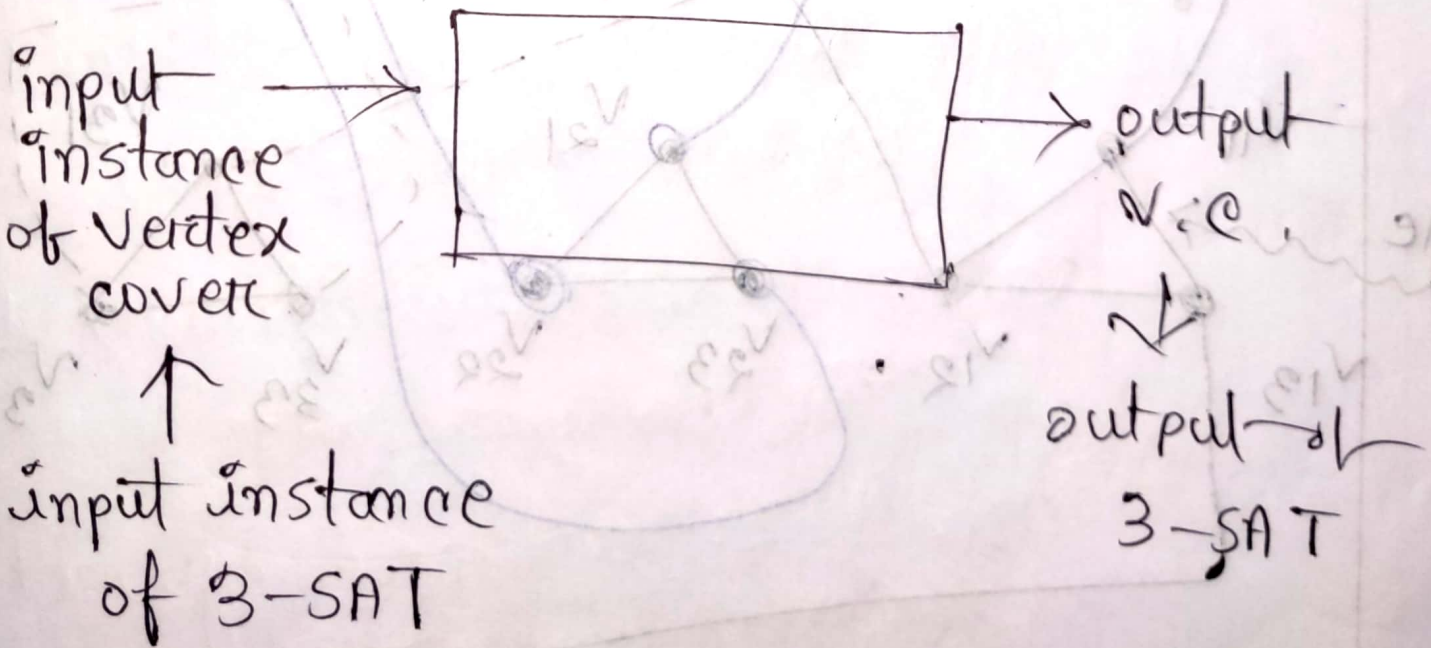
→ algo propose করতে হবে যাতে sol^n instance verify করা যায় in polynomial time.

Done in prev class.

Now, Vertex cover prob is NP-hard.

3-SAT $\leq_P$ Vertex Cover Problem

boolean ckt prob — optimization prob

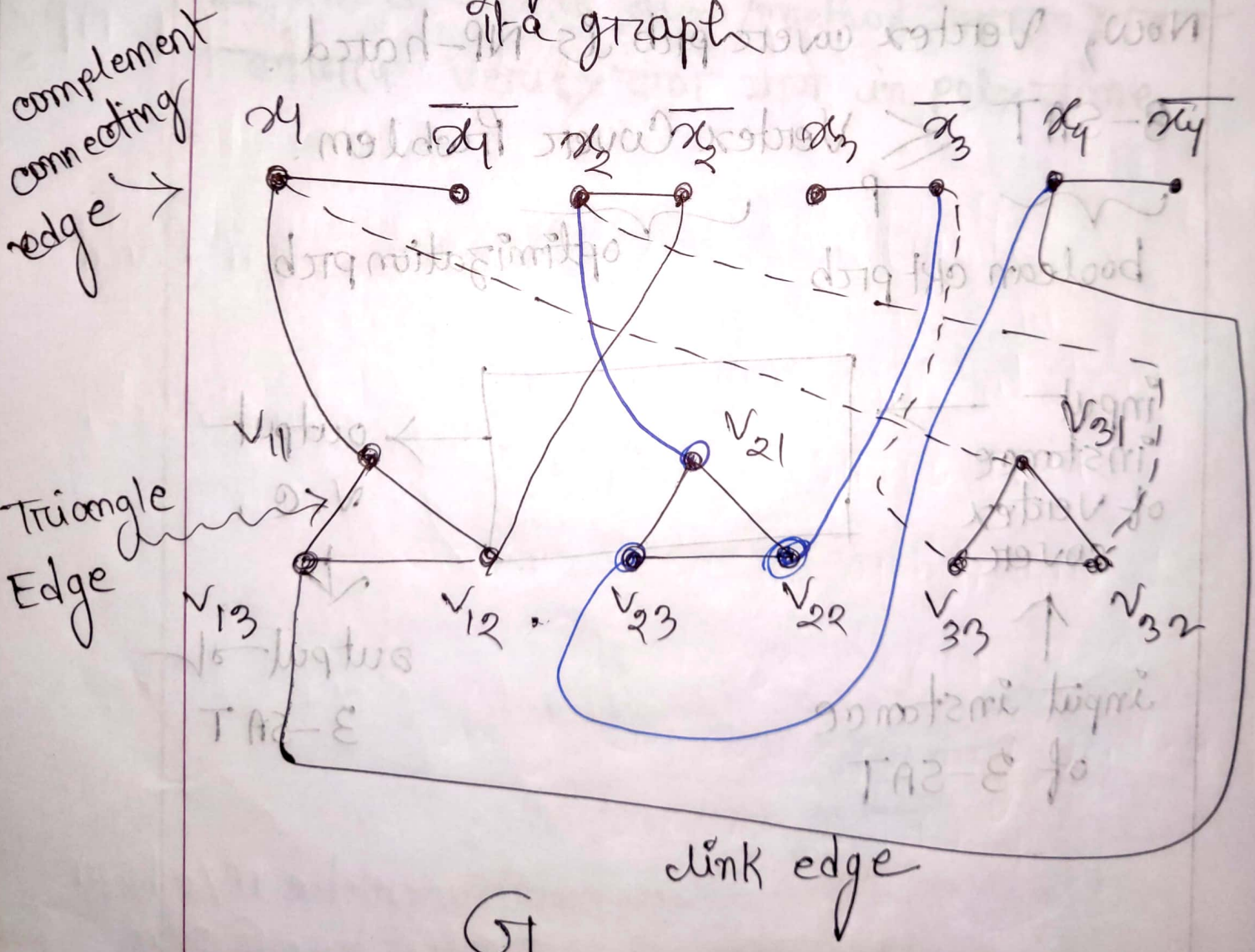


3-SAT এর যে-কোনো prob. instance..

$$Y = (x_1 + \bar{x}_2 + x_4)(x_3 + \bar{x}_3 + x_4)(x_1 + x_2 + \bar{x}_3)$$

Obs

① V.C is a graph prob. We need to make it a graph



G

② প্রতি triangle থেকে at least 2টা vertex নিতে হবে।

(3) C.C.E তে cover করতে at least 1 টা vertex নিতে হবে।

number of variables = 5

" " clause = t

∴ at least 5+2t সংখ্যক vertex থাকবে।

K = 5+2t

এটা দিয়ে কাজ হবে তার প্রমাণ কী ?

Lemma: Y is satisfiable iff G has a vertex cover of size at most K.

যেহেতু, at least K সংখ্যক vertex নিতে হবে (from construction)

∴ exactly K সংখ্যক।

Necessity:

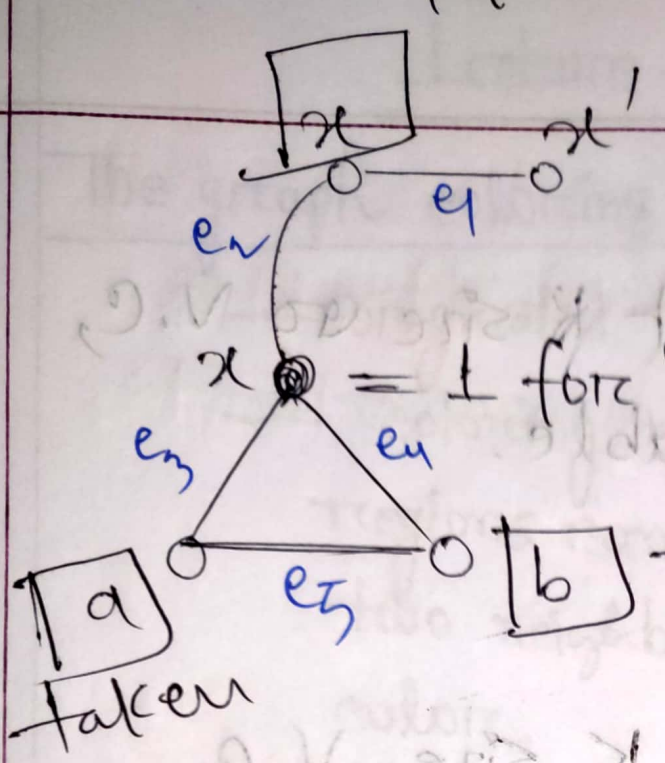
Assumption: if Y is satisfiable

তাহলে প্রতি clause থেকে at least ১টা করে 1 থাকতে হবে।

একাধিক 1 থাকলে arbitrarily 1 টা choose করবো।

এই 1 এর corresponding vertex কে vertex cover এ নিব।

from e.e.edge



e_1, e_2 covered by x
e_3, e_5 " " a
e_4, e_5 " " b

∴ x यदि 1 हय,
then,
a, b, x सब in vertex cover.

$S = \{$ যেগুলা 1 from satisfiable, আগে নিছি, $\}$
বাকি রইল vertex from triangle.

// আমরা যে flexibility দিলাম C.C.E যেখানে 1 টা vertex নেওয়া, আর triangle থেকে ২টা নেওয়া—

// এই flexibility টা মূলত exploit করলাম।

∴ S is a vertex cover.

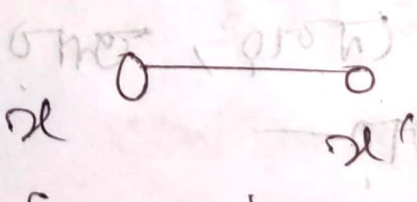
Sufficiency :

If G has at most K size এর V.C, then Y is satisfiable.

1st thing :

G has exactly K size V.C.

C.C. edge গুলা থেকে কিন্তু exactly 1 টা করে vertex V.C. তে গেসে। এমন হবেই না সে উপরে৩ region এ,



{x, x'} taken, but none from {y, y'} taken.

impossible.

∴ যখন আমাদের C.C E থেকে, তাদের 1 করে ফিব। but, but তাদের সাথের triangle এর 1টা vertex short ∴ প্রতিটা clause 1 করে।