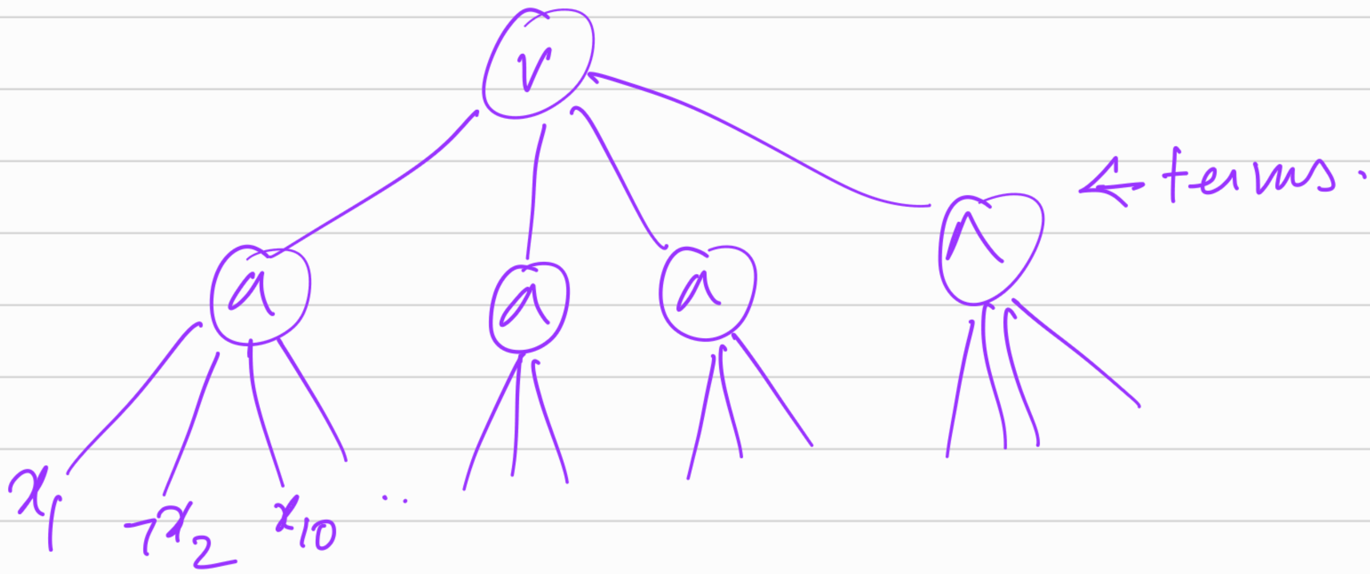
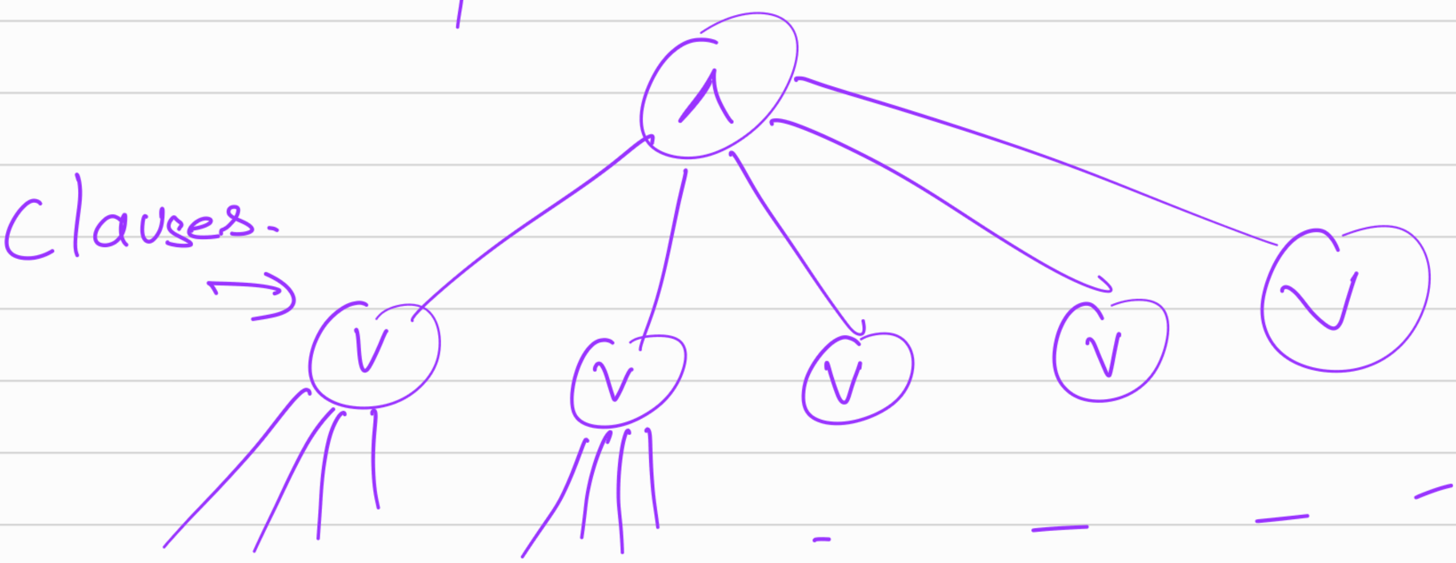
18/09/2023

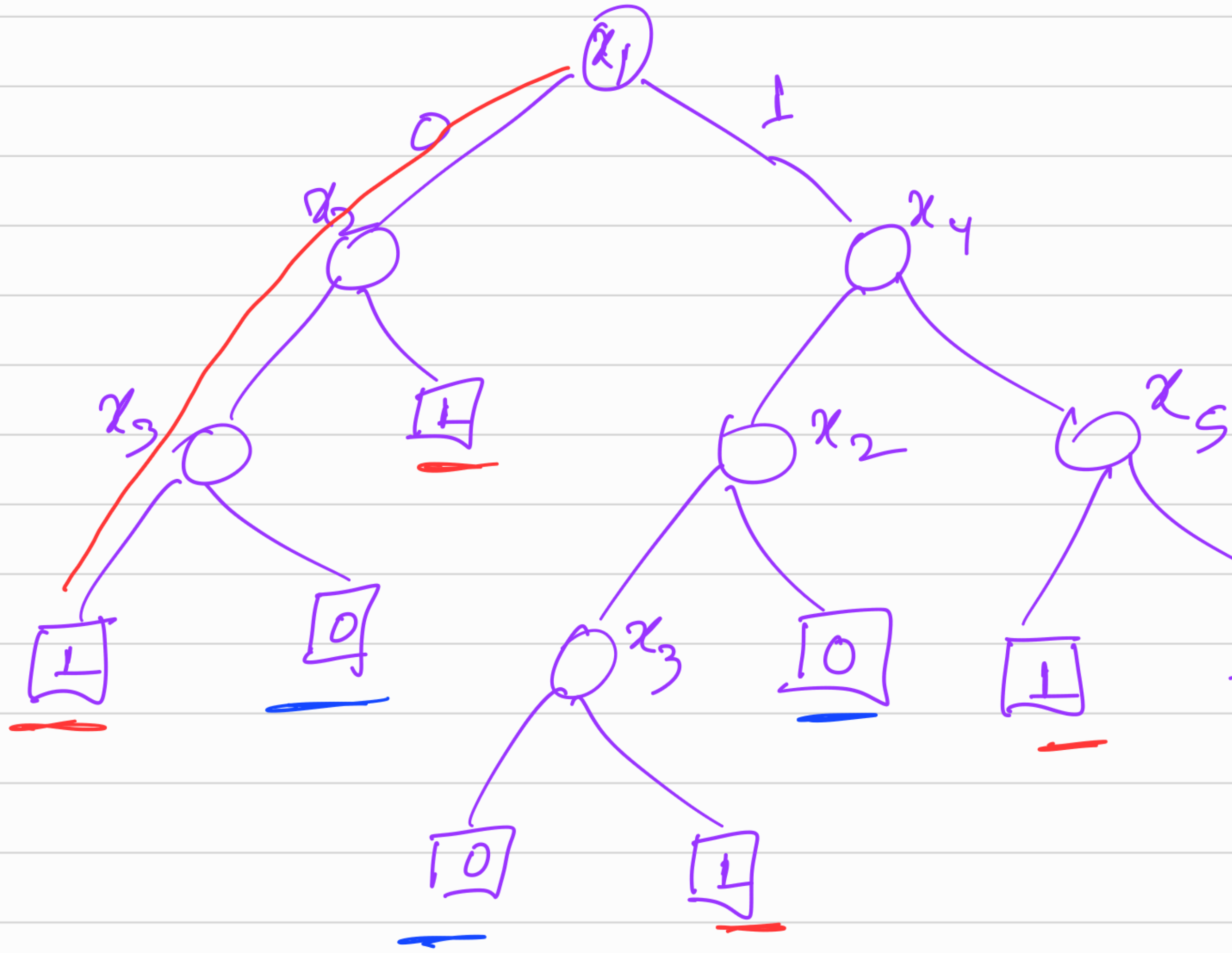
DNF - representation

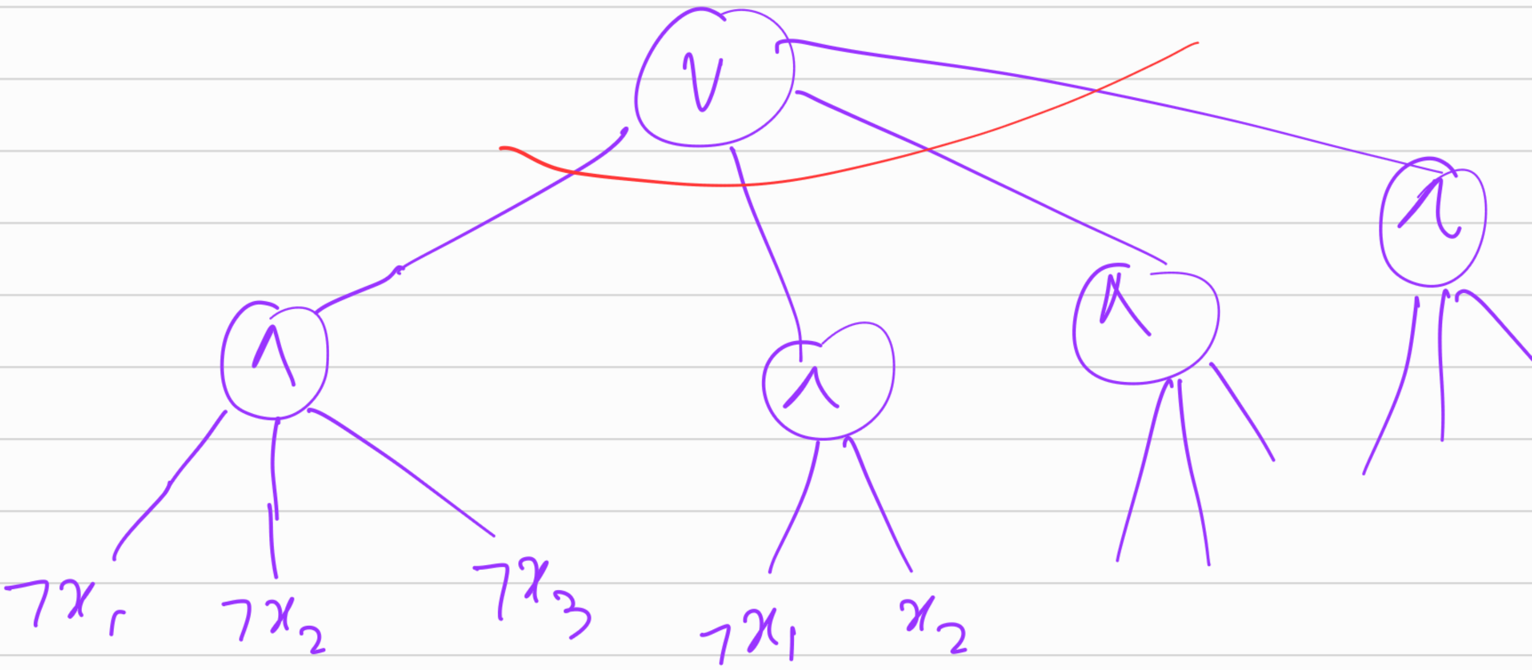


CNF - representation



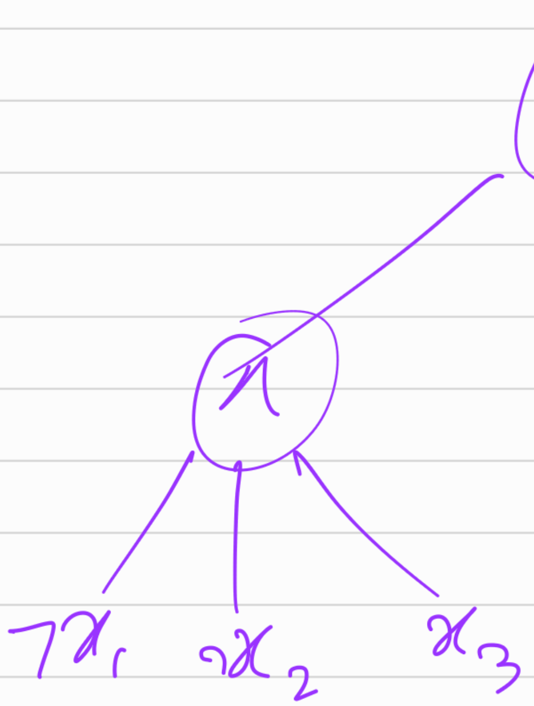
Given a DT for a function f

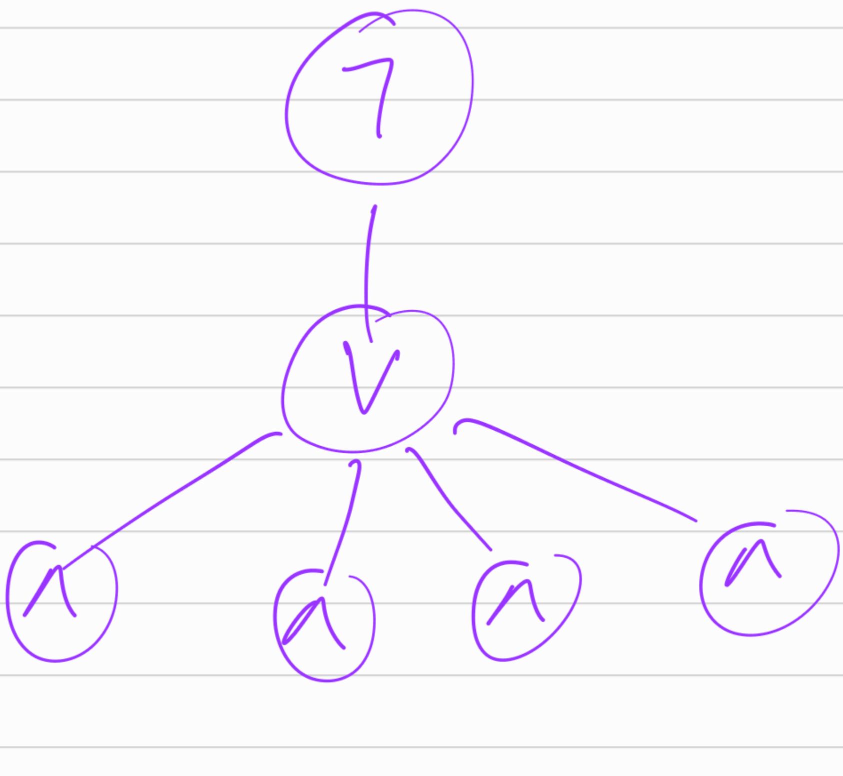




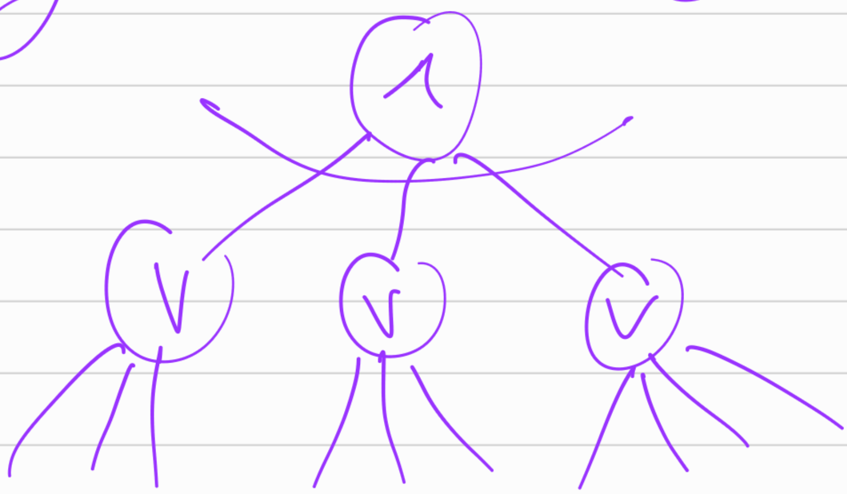
terms = # 1-leaves in DT

Using the 0-paths write a DNF for $\neg f$.





Push the negation down using De-Morgan's law



clauses = # 0-Leaves.

Obs:- If $size(DT) \leq s$

then DNF-size +

CNF-size $\leq s$

Defn:- DNF-size(f) is

the minimum # terms in a DNF representing f.

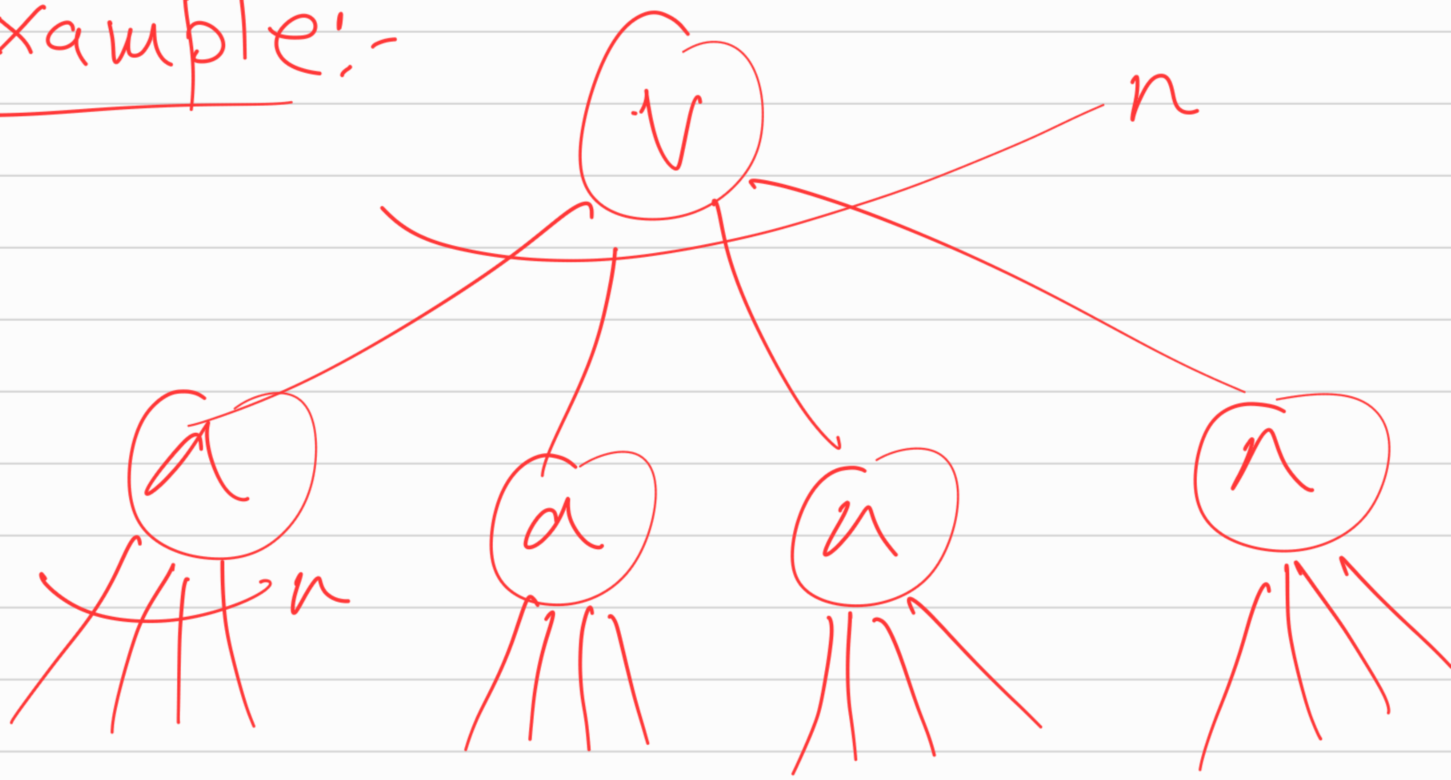
Similarly CNF-size(f).

Q: Suppose f is such that

$$\text{DNF-size}(f) + \text{CNF-size}(f) \leq N$$

then what is $L(f)$?

Example:-



Consider 0-inputs s.t. each $\wedge$ has exactly one 0.

\# such inputs $= n^n$

Q:- DNF-size(f) + CNF-size(f) $= N$.

Then what is $L(f)$?

Thm :- (Upper Bound)

Let $f : \{0,1\}^n \to \{0,1\}$

s.t. DNF-size(f) + CNF-size$(f) = N$

then $L(f) \leq n^{O(\log^2 N)}$

$= 2^{O(\log^2 N \cdot \log n)}$

Thm : (Lower Bound)

$\exists$ a function $f : \{0,1\}^n \to \{0,1\}$

s.t. DNF-size(f) + CNF-size$(f) = N$

then $L(f) \geq 2^{O(\log^2 N)}$

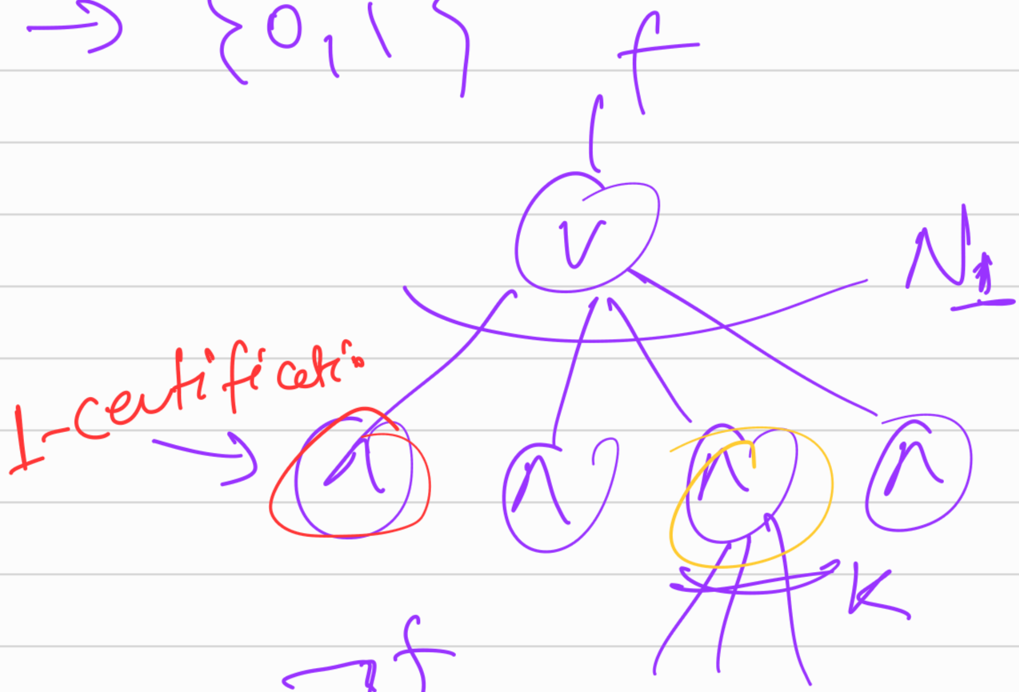
$$= N^{O(\log N)}$$

"Quasi-polynomial"

Thm: (Upper-bound)

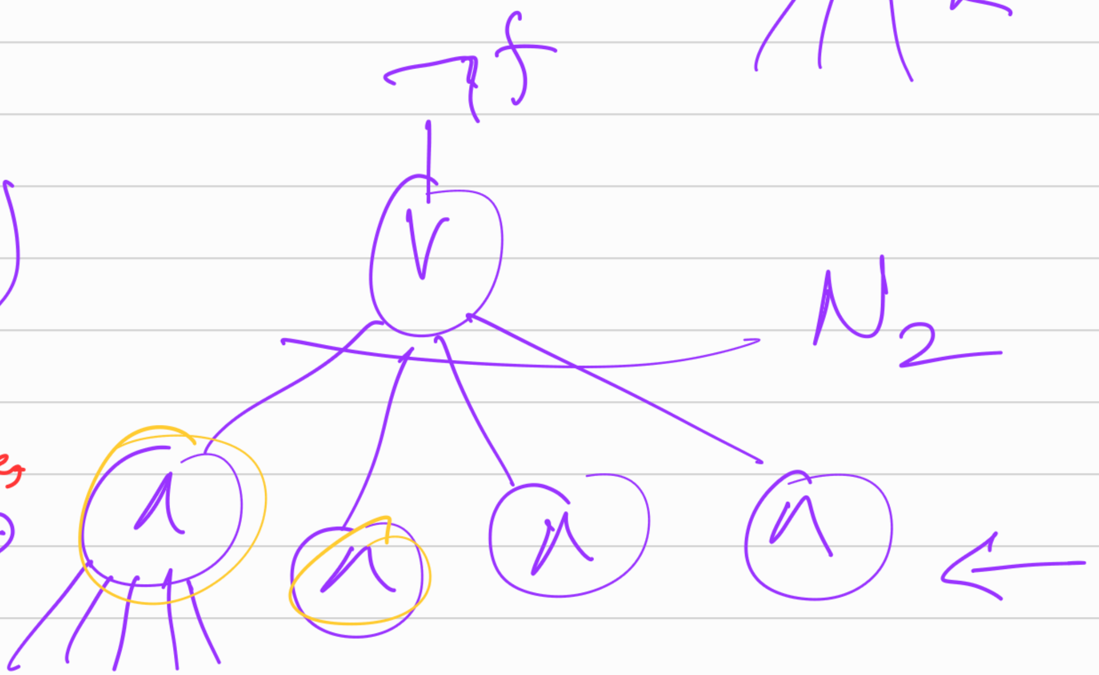
$f : \{0,1\}^n \to \{0,1\}$

DNF(f)



DNF($\neg f$)

0-certificates



$$N_1 + N_2 = N.$$

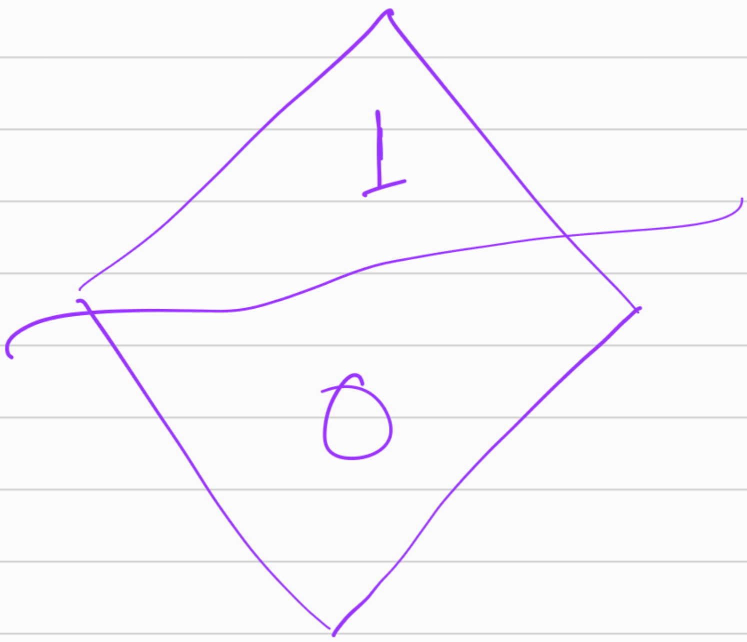
→ query a variable that appears most "often" in the two DNFs.

→ Recall 1-certificate and 0-certificate intersect in a contradictory manner.

Suppose one of the 1-certificate given in DNF is of size K.

$\Rightarrow \exists$ a variable that appears in $\frac{1}{K} \cdot N_2$ many 0-certificate.

$\rightarrow$ What is an upper bound on k?



$|\text{a certificate}| = t.$

inputs that it can cover $= 2^{n-t}$.

$2^n \leq$ sum of # inputs covered by each certificate.

$\leq N \cdot 2^{n-t}$

where t is minimum size of a certificate.

$$2^n \leq N \cdot 2^{n-t}$$

$$\Rightarrow \quad 2^t \leq N \Rightarrow t \leq \log N$$

another proof :-

In total there a N certificates that covers the whole input space.

$\Rightarrow$ $\exists$ a certificate that covers $\geq \frac{1}{N} \cdot 2^n$

$$\frac{2^n}{N} \leq 2^{n-t}$$

$\boxtimes$

Lemma :- $\exists$ a certificate of size $\leq \log N$.

DT-algorithm:

$\Rightarrow$ Pick a certificate of size $\leq \log N$.

Say this is a 1-certificate

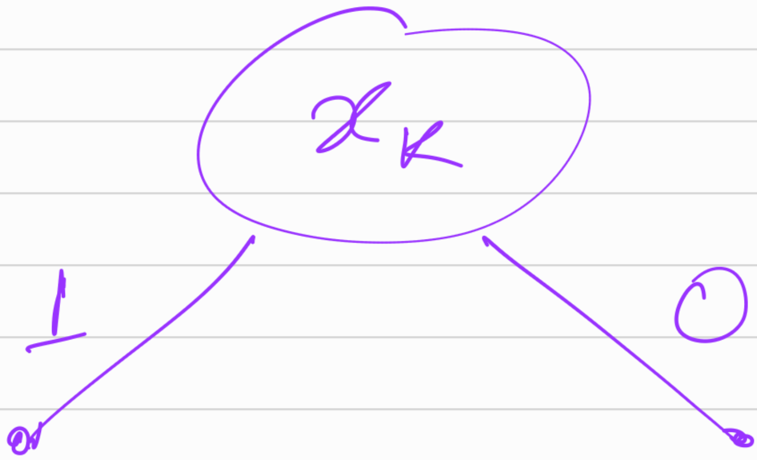
$\Rightarrow$ $\exists$ a variable that appears contradictorily in $\geq \frac{1}{\log N} \cdot N_2$ many 0-certificates.

Step 2 :- Query this variable and recurse.

Step 3 :- Output answer when one of the certificate is satisfied.

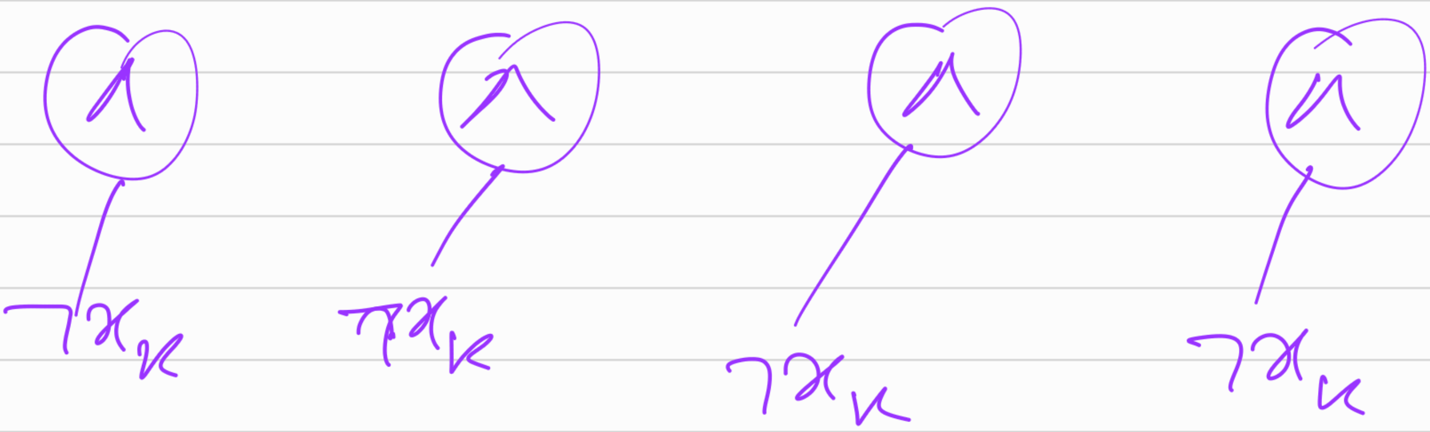
Correctness :- Obvious!

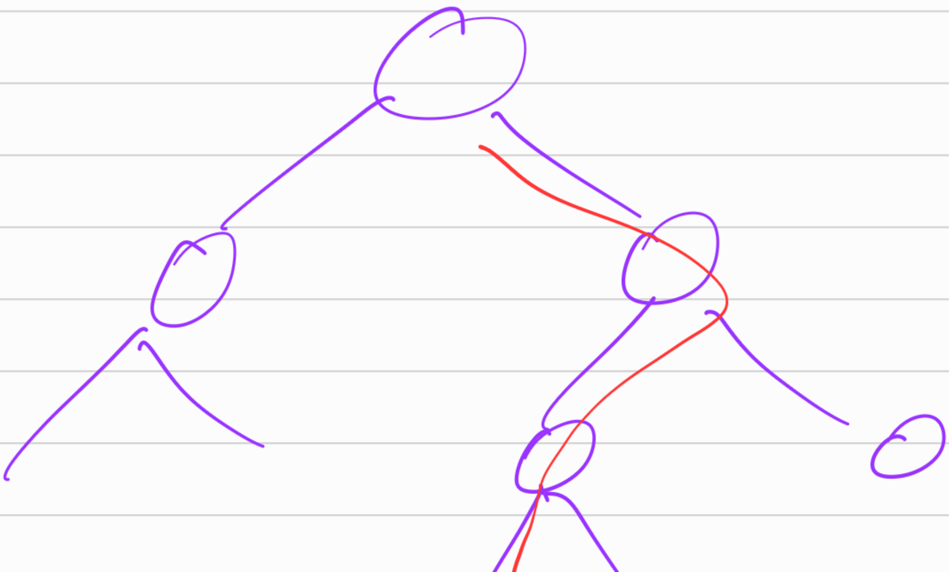
Size :-



$N_2' \leq \left(1 - \frac{1}{\log N}\right) N_2$ $\qquad$ $N_1' \leq N_1 - 1$

Suppose x_k appears positively in the 1-certificate.





Call an edge "decreasing" if following that edge kills $\geq \frac{1}{\log N}$ fraction of certificates.

Consider a path that d-decreasing edges in it.

That means one of the DNF-size or CNF-size has decreased at least $\frac{d}{2}$ times.

$$N\left(1-\frac{1}{\log N}\right)^{\frac{d}{2}} \le N \cdot e^{-\frac{1}{\log N}\cdot\frac{d}{2}}$$

$$= N \cdot e^{-\frac{d}{2\cdot\log N}}$$

(suppose $d = 2\log^2 N$):

$$\le N \cdot e^{-\log N}$$

$$\le N \cdot \frac{1}{e^{\log N}}$$

$$= N \cdot \frac{1}{N^{\log e}}$$

$$= \frac{N}{N^{1+\delta}} \quad \text{where } \delta > 0$$

$$< 1.$$

$\Rightarrow$ Any root to leaf path is this decision tree has at most $2\log^2 N$ "decreasing" edges.

Also, the maximum depth is $\leq n$.

$\Rightarrow$ label decreasing edges by 0. and non-decreasing edges by 1.

Then any root to leaf corresponds to 0-1 string of length $\underline{\leq n}$ containing $\leq 2\log^2 N$ 0s.

This # $\leq \sum_{i=0}^{2\log^2 N} \binom{m}{i}$

$$\le n^{O(\log^2 N)}$$

$$\left[\sum_{i=0}^{t} \binom{n}{i} \le n^{O(t)}\right].$$