Name :- Sumit Kumar Yadav

~~[illegible]~~ Roll No. :- 18CS30042

Page No.

Date

Solⁿ:- 1:- Type-0 grammar for language $\{0^{2^i+3^i} \mid i \geq 1\}$

consider, $G = (N = \{S, A, B, C, D, E, X, Y\},$
$\Sigma = \{0\}, P, S)$

P : ⓐ $S \rightarrow ACXYB$

ⓑ $CX \rightarrow XXC$

ⓒ $CY \rightarrow YYYC$

ⓓ $CB \rightarrow DB$

ⓔ $CB \rightarrow E$

ⓕ $XD \rightarrow DX$

ⓖ $YD \rightarrow DY$

ⓗ $AD \rightarrow AC$

ⓘ $XE \rightarrow E0$

ⓙ $YE \rightarrow E0$

ⓚ $AE \rightarrow \varepsilon$

$\therefore$ X generates strings $0^{2^i}$ $\Rightarrow$ y generates $0^{3^i}$.

Page No.
Date

Solⁿ. 2:-(a) Hardness:-

The language $L'$ is hard if $L \leq_m L'$ for every $L \in$ r.e

Completeness:-

The language $L'$ is r.e. complete if $L'$ is r.e hard and $L' \in$ r.e.

(b) Example ~~of~~ for an r.e complete language:-

halting problems:-

because
↳ it is recursively enumerable
↳ is r.e. hard

(c) language that is r.e. hard but not r.e complete

⇒ let $INFIN = \{\langle M \rangle \mid L(M) \text{ is infinite}\}$

now, we can show that $L_\epsilon \leq INFIN$

$$(M, x) \mapsto N$$

N: on any input y,
→ if $|y| \leq |x|$ accepts y
→ else run M on x, if M accepts x, accept y.

$$\therefore L(N) = \begin{cases} \overset{\text{finite}}{\{s\}} \mid s \in \Sigma^* \quad |s| \leq |x| & \text{if } M \text{ doesn't halt on } x \\ \Sigma^* & \text{if } M \text{ accepts } x \end{cases}$$

Hence, INFIN is r.e. hard
(FIN & INFIN do not belong to r.e. sets)

Page No.

Date

Solⁿ: 3:- Consider G: graph.

$P \Rightarrow$ length of path betⁿ. s & t

now, consider the path as a sequence of edges $e_1, e_2, \ldots, e_p$

s.t. $e_1 = (s, u)$, $e_p = (v, t)$

where $u, v \in V$

now for $i$: 1 to $P$

→ delete $e_i$

→ run dijkstra to find the new distances

→ check the shortest path found to t, if length $\leq 100k$, if solved then return yes.

otherwise continue.

→ when $i > P$ return no as no such path found.

∴ Clearly, this problem would take polynomial time to run, therefore so next-path is in P.

Page No.
Date

Soln:- 4:- $\because$ non trivial SAT problems takes an input a CNF-SAT formula $\phi$.

$\therefore$ non trivial SAT is in NP as given in question of true/false of each literal,

$\therefore$ we can check whether formula is satisfyable and if each clause has a true literal and a false literal,

Now To show NP hardness:-

we reduce Non trivial SAT from 3 SAT

$$3\,SAT \leq \text{Non-trivial SAT}$$

$\therefore$ given instance of 3-SAT, $\phi$ for each clause

$C_i \in P$ of form $C_i = (x_i \vee x_j \vee x_k)$

we can construct $C_i' = (y_i \vee y_j \vee y_k \vee z)$

where z is common variable & for every variable $x_i$ there is corresponding variable $y_i$.

So, $C_i'$ represent $\phi'$ instance of non trivial SAT.

$\therefore$ $P \in$ 3-SAT $\Leftrightarrow$ $P' \in$ Non-Trivial SAT

as $C_i' = (y_i \vee y_j \vee y_k \vee z)$ is true

& if and only if atleast one is true & one is false.

$\therefore$ $x_i = y_i \oplus z$

$\therefore$ we can say that the,

Non-trivial SAT is NP complete.