AMIT YADAV

HW

H1.1

(i) To Prove : for $r, s > 1$

$$(\log_2 n)^s = o(n^r)$$

Solution :

To prove that $(\log_2 n)^s = o(n^r)$, it is sufficient to prove that $\log_2((\log_2 n)^s) = o(\log_2 n^r)$ since, 'log' is a monotonically increasing function.

Now,

$$\log_2((\log_2 n)^s) = s \log_2(\log_2 n) \quad \text{—— (I)}$$

$\cancel{<}$ $\cancel{s \log_2 n}$ $\quad$ $\because$ $\cancel{\log_2 n} < \cancel{n}$ $\cancel{\forall} n > 0$

and

$$\log_2 n^r = r \log_2 n \quad \text{—— (II)}$$

Checking the limit :

$$\lim_{n \to \infty} \frac{s \log_2(\log_2 n)}{r \log_2 n}$$

$$= \lim_{n \to \infty} \frac{\frac{s}{r} \times \frac{1}{\log_2 n} \times \frac{1}{n}}{\frac{1}{n}} \quad \text{(Using L'Hôpital rule)}$$

$$= \lim_{n \to \infty} \frac{s}{r} \times \frac{1}{\log_2 n} = 0$$

$\therefore$ We can say that $\log_2((\log_2 n)^s) = o(\log_2 n^r)$

Hence $(\log_2 n)^s = o(n^r)$.

H1.1

(ii) To show: $\log_2 n! = \sum_{i=1}^{n} \log_2 i = \Theta(n \log_2 n)$

Solution:

$$\sum_{i=1}^{n} \log_2 i \le \sum_{i=1}^{n} \log_2 n \qquad \forall\, n > 0$$

$$= n \log_2 n$$

$$\therefore \sum_{i=1}^{n} \log_2 i = O(n \log_2 n) \quad \text{—— } (1)$$

Also,

$$\sum_{i=1}^{n} \log_2 i \ge \overbrace{\sum_{i=n/2}^{n} \log_2 i}^{\text{Last } n/2 \text{ terms}} + \overbrace{\sum_{i=1}^{n/2} \log_2 2}^{\text{first } n/2 \text{ terms}} \quad [\forall\, n > 0]$$

$$\ge \sum_{i=n/2}^{n} \log_2 (n/2) + \frac{n}{2}$$

$$= \frac{n}{2} \log_2 \left(\frac{n}{2}\right) + \frac{n}{2}$$

$$= \frac{n}{2} \log n - \frac{n}{2} + \frac{n}{2}$$

$$= \frac{n}{2} \log_2 n$$

$$\therefore \sum_{i=1}^{n} \log_2 i = \Omega(n \log_2 n)$$

since $\sum_{i=1}^{n} \log_2 i \ge c \cdot n \log_2 n$, for $c = \frac{1}{2}$ and all $n > 0$.

$$\therefore \sum_{i=1}^{n} \log_2 i = \Theta(n \log_2 n).$$

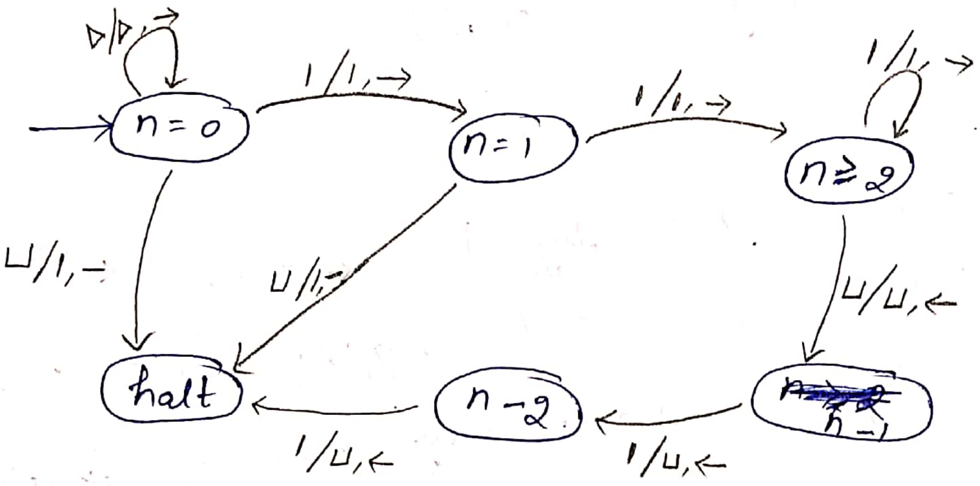
Scanned by CamScanner

H1. 2

$$f(x) = \begin{cases} 1^{n-2} & \text{if } x = 1^n \text{ and } n \geq 2 \\ 1^{n+1} & \text{if } x = 1^n \text{ and } n < 2 \end{cases}$$

Turing machine:

Graphical representation:



$M = (K, \Sigma, \delta, s)$

Here, $\Sigma = (\sqcup, \triangleright, 1)$

$s = (n=0)$

Computation sequence for input

(i) 11 : $(n=0, \triangleright, 11)$

$(n=0, \triangleright 1, 1\underline{1})$

$(n=1, \triangleright 11, \sqcup)$

$(n=2, \triangleright 11\underline{\sqcup}, \sqcup)$

$(n-1, \triangleright 11, \sqcup)$

$(n-2, \triangleright 1, \sqcup\sqcup)$

$(halt, \triangleright, \sqcup\sqcup\sqcup)$

Hence $M(11) = \varepsilon$

(ii) ε :

$(n=0, \triangleright, \sqcup\sqcup ..)$

$(n=0, \triangleright\sqcup, \sqcup\sqcup ...)$

$(halt, \triangleright 1, \sqcup\sqcup)$

H1.3

(i) Let $\exists M$ a TM, such that M decides L.

Then, $\exists M'$, such that M' ~~dedi~~ decides $\bar{L}$.

Simply exchange the states named "yes" and "no".

So, if M outputs "yes" on x, then M' will output "no" and vice-versa.

So, M' exists.

This argument doesnot hold for accepting languages.

Because, let's say there is a string x on which M never halts. Then $x \notin L(M)$.

But there might not exist a TM M' which accepts x.

So this argument doesn't work for accepting languages.

Q1.3

(ii) TM M_1 accepts $L \subseteq \{0,1\}^*$

TM M_2 accepts $\bar{L} \subseteq \{0,1\}^*$

To show: $\exists M$ such that M decides L.

<u>Solution</u>:

① Run M_1 and M_2 in parallel, on any given string.

whichever occurs first

② If M_1 accepts the string, then accept it.

$\boxed{\text{OR}}$

③ If M_2 accepts the string, then reject it.

Since, one of M_1 or M_2 must accept a given string, ~~so~~,

so 'L' is decidable.

Scanned by CamScanner