- Review oracle machines, SAT vs. 3SAT and SUBSET-SUM "solution"

Oracle T.m. :

Turing machine, M, language L; we write M^L = machine M with oracle L:

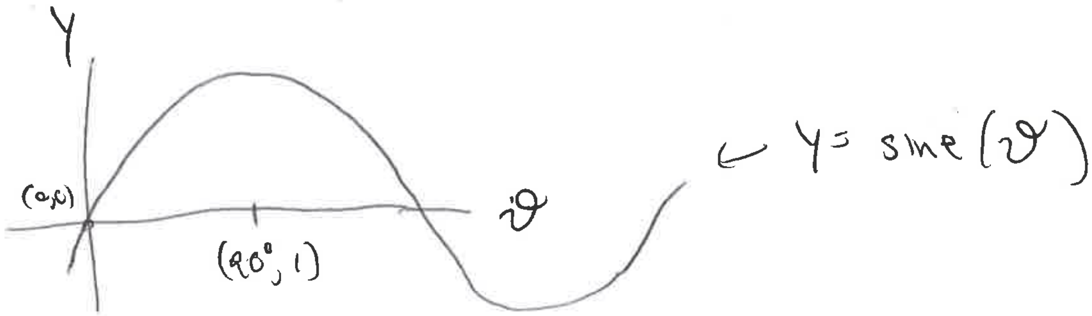
usual T.M setup + $q_{\text{query } L}$ + $q_{\substack{\text{oracle} \\ \text{yes}}}$ + $q_{\substack{\text{oracle} \\ \text{no}}}$

Enter $q_{\text{query } L}$ $\longrightarrow$ $\begin{cases} \text{if word on tape 1 is in } L \to q_{\substack{\text{oracle} \\ \text{yes}}} \\ \text{-- -- -- -- not in } L \to q_{\substack{\text{oracle} \\ \text{no}}} \end{cases}$

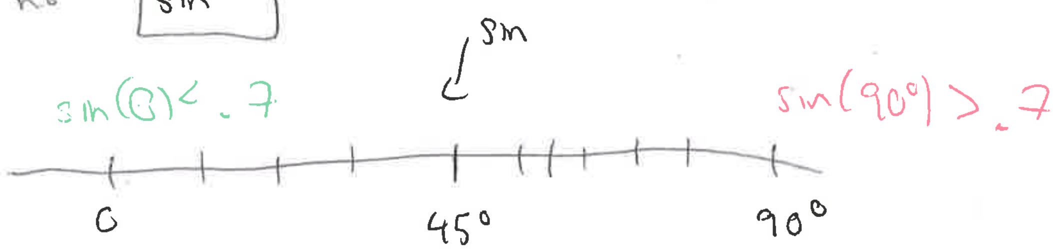
Like you have button on your calculator that you press to test membership in L.

Reduction: $L_1 \leq_p L_2$ means you run a poly-time alg and you get to test membership in L_2 once.

What's the difference ?



Say you solve $\sin(\theta) = .7$. Say you have $\boxed{\sin}$ but no $\boxed{\sin^{-1}}$



etc.

If you have one $\boxed{\sin}$ call ☹
— — — a lot $\boxed{\sin}$ call (or one $\boxed{\sin^{-1}}$) ☺

Function $f_n : \Sigma^n \to \Sigma^n$ for all n (f really $f_1, f_2, \ldots$)

s.t.
1. f_n is a bijection
2. f_n is "easy" to compute
3. f_n^{-1} is not easy to compute

} "One-way functions"

Algorithm: That for each $w \in \Sigma^n$ can find $x_1, \ldots, x_{n^3} \in \Sigma^n$ s.t. $f^{-1}(w)$ is one of $x_1, \ldots, x_{n^3}$ ...

Conceivable that a lot of calls to an "oracle" is better than one call to it at the very end.

So reduction:

Cook-Levin: If L_1 is recognized by a TM non-det in time n^k, then

$$f: \Sigma_1^* \longrightarrow \Sigma_2^* \quad \left(\begin{matrix}\text{computable} \\ \text{in } P\end{matrix}\right) \quad \left(\begin{matrix}\Sigma_1 \text{ alphabet} \\ \text{of } L_1\end{matrix}\right)$$

where $\Sigma_2 = \{ x, (,), 0, \ldots 9, \text{and}, \text{or}, \text{not} \}$ s.t.

$$\forall w \in \Sigma_1^*, \quad w \in L_1 \iff f(w) \in SAT.$$

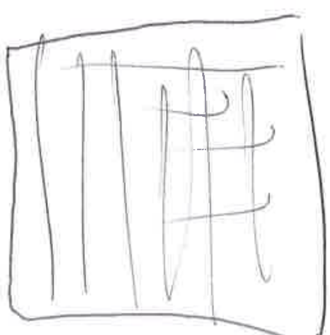
(We write $L_1 \leq_p SAT$)

The Cook-Levin theorem also work for 3SAT:

$$3SAT = \left\{ \langle f \rangle \;\middle|\; \begin{matrix} f \text{ is a Boolean function } 3CNF \text{ form} \\ \text{s.t. } f \text{ is satisfiable} \end{matrix} \right\}$$

Cook-Levin

n^k steps



n^k cells

there's an accepting comp

iff

( ) AND ( ) AND ( ) ...

(blah_1) AND
(blah_2) AND
(blah_3) AND
.
.
$(\text{blah}_{O(n^{2k})})$

3CNF: ( ) AND ( ) AND ... AND ( )

x_i or x_j or x_k

x_i or $\neg x_j$ or $\neg x_k$

Main tricks : SAT $\rightsquigarrow$ 3SAT :

"rewrite"

$(x_1$ or x_2 or x_3 or x_4 or $x_5)$ $\qquad x_{n^4}, \ldots$

$($ $\quad$ $)$ AND $($ $\quad$ $)$ - - ~

or or $\qquad$ or or

Trick : we'll do this but add variables : add $y_1, y_2, \ldots$

$(x_1$ or x_2 or x_3 or $x_4)$ is true iff

$(x_1$ or x_2 or $y_1)$ AND $(\neg y_1$ or x_3 or $x_4)$ is satisfiable

$(x_1$ or $\ldots$ $x_5)$ similar, but ~~need~~ to add y_1 ~~and~~ y_2

Midterm :

5

Pile 1: A-H

Pile 2: J-L

Pile 3: M-Q

Pile 4: R-V

Pile 5: W-Z

Linear Interpolation

$0 \longrightarrow 0\%$

Scaling:

$15/50 \xrightarrow{\text{scales to}} 50\%$

$35/50 \xrightarrow{\text{scales to}} 80\%$

$47/50 \xrightarrow{\text{scales to}} 100\%$

Median = $26.5/50$