Lesson 23

Sunday, November 25, 2018 6:05 PM

I. HW #5 questions
(go over all).

II. NP:

$$\text{CLIQUE} = \{ \langle G, k \rangle \mid G \text{ is a graph that contain a } k\text{-clique} \}$$

Recall: a k-clique is a set $C \subseteq V$ s.t. $|C| = k$ and all vertices in C have edges between them.

* Show that CLIQUE has a poly. time verifier.

* Exercise: 3-CLIQUE $\in P$!

Turns out CLIQUE is NP-complete.

III Decision vs. search problems.

Decision problem: given $\langle G, k\rangle$ determine if G has a k-clique.

Search problem: given $\langle G, k\rangle$, find a subset $C \subseteq V$ s.t. C is a k-clique.

<u>Ex:</u> Suppose CLIQUE $\in$ P. Show that there is a polyn. time alg. which, given a graph G and a # k, outputs a k-clique or outputs "No k-clique" if there is none.

Given $G = (V, E), k$:
$= (v_1, \ldots, v_n), (e_1, \ldots, e_\ell), k$

1. Check if $\langle G, k\rangle \in$ CLIQUE.
   If not, output "No k-clique."

set $\tilde{G} = G$
   for $i = 1$ to n:
      Form G_i by deleting v_i from $\tilde{G}$.
      If $\langle G_i, k\rangle \in$ CLIQUE,
         set $\tilde{G} = G_i$ and continue

If $\langle G_i, k \rangle \in$ CLIQUE,
set $\tilde{G} = G_i$ and Continue.

⋮

Output any k nodes of $\tilde{G}$.

Idea: $G_0 = G$

$$G_1 = \begin{cases} G \setminus v_1 & \text{if that has a } k\text{-clique} \\ G & \text{o/w} \end{cases}$$

ie: $v_1 \in G_1$ iff all k-cliques in G contain v_1.

$$G_2 = \begin{cases} G_1 \setminus v_2 & \text{if that has a } k\text{-clique} \\ G_1, & \text{o/w.} \end{cases}$$

$v_2 \in G_2$ iff all k-cliques in G_2 contain v_2.

ie: $v \in \tilde{G}$ iff all k-cliques in G contain v.

HW: Similar, for SUBSET-SUM.

(IV) Turing jumps.

Recall:

① $f: \mathbb{N}^k \to \mathbb{N}$ is partial computable if...

(Universal TM)

② There is $f: \mathbb{N}^2 \to \mathbb{N}$ partial computable such that, for all partial computable $\varphi: \mathbb{N} \to \mathbb{N}$, there is $e \in \mathbb{N}$ such that for all $n \in \mathbb{N}$ "$\varphi(n) = f(e, n)$",

ie, if $\varphi(n)$ doesn't halt, neither does $f(e,n)$, and if it does, so does $f(e,n)$ and they both output the same #.

We write f_e to refer to $f(e, \cdot)$. Then $f_0, f_1, f_2, \ldots$ is a computable enumeration of all partial computable functions.

- Write $f(n)\downarrow$ to mean $f(n)$ halts.

- Write $f(n)\downarrow$ to mean $f(n)$ halts.

  $f(x)\downarrow = y$ means it halts and outputs y.

  $f(n)\uparrow$ means it does not halt.

A set $A \subseteq \mathbb{N}$ is computable if its char. function χ_A is

ie: $\chi_A(x) = \begin{cases} 0, & \text{if } x \notin A \\ 1, & \text{if } x \in A. \end{cases}$

$X = \{e \mid f_e(e)\downarrow = 0\}$ is not computable.

Pf:

Let n be s.t. $\chi_X = f_n$.

Q: • $f_n(n) = 0$?
Then $n \in X$. But then $\chi_X(n) = 1$!

• $f_n(n) = 1$? Then $\chi_X(n) = 1$,
so $n \in X \Rightarrow f_n(n)\downarrow = 0$?

Contradiction either way.

<u>Cor</u>:

$H = \{ e \mid f_e(e) \downarrow \}$ is not computable.

Pf: $X \leq_T H$. If H is computable,

Then $\chi_X(n)$:

$n \notin H \to$ output 0.

$n \in H$: compute $f_n(n)$. If $f_n(n) = 0$, output 1, o/w, output 0.

<u>Def</u>: Let $A \subseteq \mathbb{N}$. A (partial) function f is <u>A-computable</u> if there is an oracle TM, allowing access to an <u>oracle</u> for A, which computes f as before.

<u>Recall</u>: oracle lets you ask: "is $x \in A$? is $x \notin A$?"

Given X, H as above:

χ_X is H-computable.

We say A is B-computable if χ_A is B-computable.

We say A is B computable if χ_A is B-computable.

" $A \leq_T B.$ "

Fact: There is a "universal oracle-TM" Φ, similar to the "universal TM".

ie: for each $X \subseteq \mathbb{N}$, $\Phi_e^X(n)$ is a partial X-comp. fn s.t. $\forall$ partial X-comp $\varphi^X : \mathbb{N} \to \mathbb{N}$, there is e so that

$$\Phi_e^X(n) = \varphi(n)$$

Def: Let $A \subseteq \mathbb{N}$. Then the <u>Turing jump</u> of A, denoted A', is the halting problem relativized to A:

$$\{e \mid \Phi_e^A(e) \downarrow\}.$$

Claim: A < A'

Claim: $A <_T A'$.

That is, A' is not A-computable.

(Same proof that H is not computable).

$\emptyset$ - computable sets: computable sets.

$\emptyset'$: c.e. or co-c.e. sets.

$\emptyset''$: like FIN, TOT, EQ_{TM}.

$\emptyset'''$?

⋮

There is no set $X \subseteq \mathbb{N}$ such that for all $A \subseteq \mathbb{N}$, $A \leq_T X$.