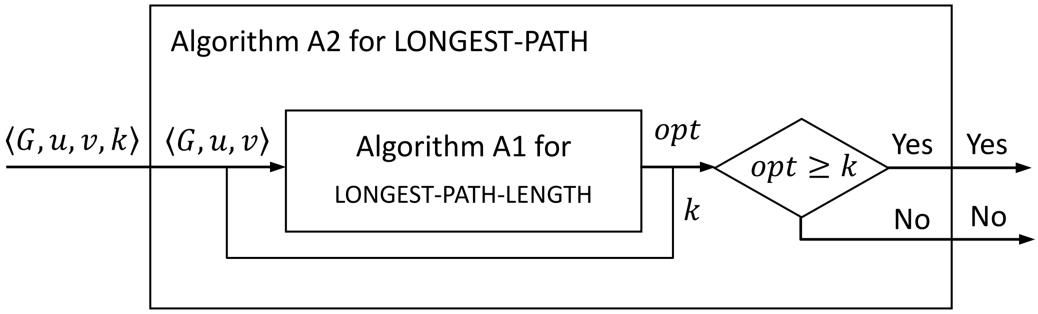
HW#5 solution

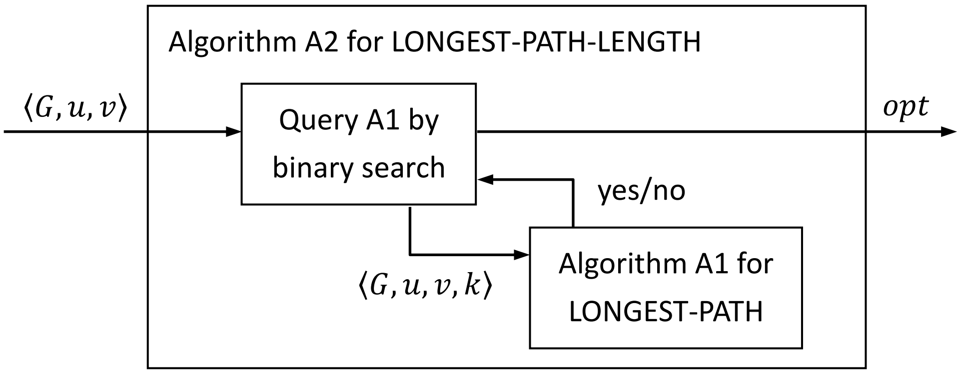
1 Ex. 34.1-1

(⇒) The decision problem LONGEST-PATH may be reduced to the optimization problem LONGEST-PATH-LENGTH as follows:



Since, by assumption, LONGEST-PATH-LENGTH can be solved in polynomial time, Algorithm A1 takes a time in $O(|\langle G, u, v\rangle|^c)$ for some constant $c > 0$.
Since $|\langle G, u, v\rangle| < |\langle G, u, v, k\rangle|$, Algorithm A2 takes a time in
$O(|\langle G, u, v\rangle|^c) + O(1) = O(|\langle G, u, v, k\rangle|^c)$
which is of polynomial time in terms of the input size $|\langle G, u, v, k\rangle|$.
Thus, LONGEST-PATH ∈ P

(⇐) The optimization problem LONGEST-PATH-LENGTH may be reduced to the decision problem LONGEST-PATH as follows:



Algorithm A2 uses binary search to query algorithm A1 as follows:

Step 1: Set $\min = 0, \max = n$ where $n =$ the number of vertices in G
Step 2: If $\max - \min = 1$ then set $opt = \min$ and terminate
Step 3: Call algorithm A1 with $k = \lfloor(\min + \max)/2\rfloor$
Step 4: If algorithm A1 answers yes, set $\min = k$ else set $\max = k$
Step 5: Goto step 2

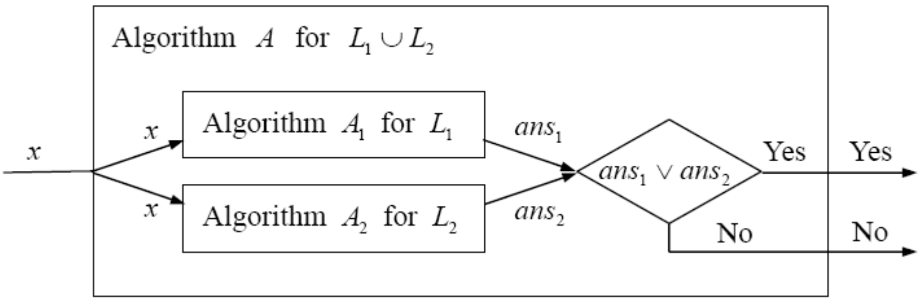
Observe that the invariant of the binary search is
min $\leq$ optimal solution $<$ max
Hence, if max $-$ min $= 1$, the optimal solution equals to min.

Since, by assumption, LONGEST-PATH $\in$ P, Algorithm A1 takes a time in $O(|\langle G, u, v, k\rangle|^c)$ for some constant $c > 0$
Cleary, Algorithm A1 is called $\lg n$ times, each time with a different value of k.
Since the value of k is always less than n, algorithm A2 takes a time in $O(\lg n \cdot |\langle G, u, v, n\rangle|^c)$.
Since $n \leq |\langle G, u, v\rangle|$ ($\because$ the encoding $\langle G, u, v\rangle$ contains at least n bits) and $|\langle n\rangle| = \lg n \leq n$ (because of standard binary encoding), it follows that
$|\langle G, u, v, n\rangle| = |\langle G, u, v\rangle| + |\langle n\rangle| \leq 2 \cdot |\langle G, u, v\rangle|$
Hence, algorithm A2 takes a time in
$O(\lg n \cdot |\langle G, u, v, n\rangle|^c) = O(\lg|\langle G, u, v\rangle| \cdot (2 \cdot |\langle G, u, v\rangle|)^c)$
which is of polynomial time in terms of the input size $|\langle G, u, v\rangle|$.

2 If $L_1, L_2 \in P$, then $L_1 \cup L_2 \in P$
Let A_1 and A_2 be the polynomial-time algorithms that decide L_1 and L_2 in $O(|x|^c)$ and $O(|x|^k)$ time, respectively.
Then, the algorithm A depicted below runs in

$$O(|x|^c) + O(|x|^k) + O(1) = O\left(|x|^{\max(c,k)}\right)$$

time and decides $L_1 \cup L_2$



3 Ex. 34.3-2

Since $L_1 \le_p L_2$ and $L_2 \le_p L_3$, there are polynomial-time computable functions f and g such that $x \in L_1 \Leftrightarrow f(x) \in L_2$ and $x \in L_2 \Leftrightarrow g(x) \in L_3$.

To see that $L_1 \le_p L_3$, let $h = g \circ f$.

Then,

$x \in L_1 \Leftrightarrow f(x) \in L_2 \Leftrightarrow g\big(f(x)\big) = h(x) \in L_3$

Furthermore, suppose the computations of $f(x)$ and $g(x)$ take $O(|x|^c)$ and $O(|x|^k)$ time, respectively. Then, the computation of $h(x)$ takes a time in

$O(|x|^c) + O(|f(x)|^k)$

$= O(|x|^c) + O(|x|^{ck}) \because |f(x)| = O(|x|^c)$

$= O\big(|x|^{\max(c,ck)}\big)$

4 Ex. 34.3-6

We shall prove this theorem:

THEOREM Every language, except for $\emptyset$ and $\{0,1\}^*$, in P is P-complete.

Proof

Let $L \in P, L \neq \emptyset$ and $L \neq \{0,1\}^*$

We show that $L' \le_p L$ for every $L' \in$ P.

Since $L \neq \emptyset$ and $L \neq \{0,1\}^*$, there exist two instances $x_{yes} \in L$ and $x_{no} \notin L$.

We define the reduction function f as follows:

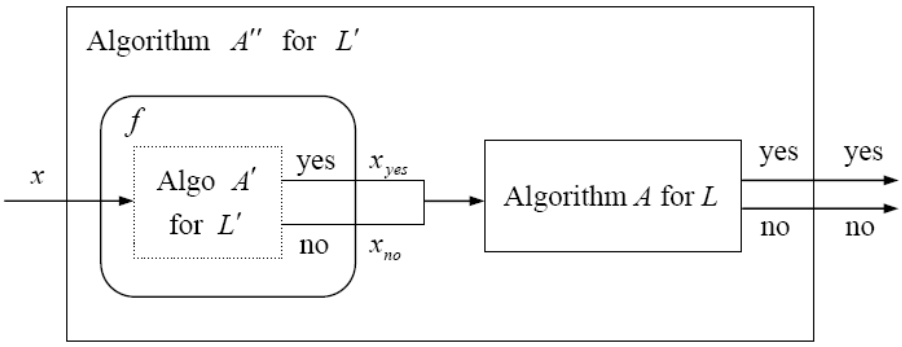
$$f(x) = \begin{cases} x_{yes} & \text{if } x \in L' \\ x_{no} & \text{if } x \notin L' \end{cases}$$

Clearly, $x \in L'$ if and only if $f(x) \in L$

Furthermore, since $L' \in$ P, the function f is polynomial-time computable.

Remarks

1 What we have proven may be depicted as follows.



The proof is a little bit tricky – we use an existing algorithm A' for L' to construct another algorithm A'' for L'.

2 $\emptyset$ and $\{0,1\}^*$ are not P-complete, because there is no yes-instance x_{yes} in $\emptyset$ and no no-instance x_{no} in $\{0,1\}^*$. Thus, $\emptyset$ and $\{0,1\}^*$ are easier than other problems in P.

3 What this theorem says is that all problems, except for $\emptyset$ and $\{0,1\}^*$, in P are equally hard (or easy).

5 Ex. 34.4-4

Let TAUT be the problem in question.

By Ex.34.3-7, TAUT is co-NPC if and only if its complement is NPC.

Thus, we need only show

THEOREM $\overline{\text{TAUT}} \equiv \{\langle\phi\rangle \mid \text{the boolean formula } \phi \text{ isn't a tautology}\}$ is NPC

Proof

$\overline{\text{TAUT}} \in \text{NP}$

Given a truth assignment for a formula that is not a tautology, we may verify whether the formula evaluates to 0 under the truth assignment in linear time by simply replacing each variable in the formula by the corresponding value and then evaluating the resulting formula.

Next, we show that $\text{SAT} \leq_{\text{p}} \overline{\text{TAUT}}$

The required reduction function is $f(\phi) = \neg\phi$

Clearly, f may be computed in polynomial time.

Furthermore,

ϕ is satisfiable

$\Leftrightarrow$ ϕ evaluates to 1 under some truth assignment

$\Leftrightarrow$ $\neg\phi$ evaluates to 0 under some truth assignment

$\Leftrightarrow$ $\neg\phi$ is not a tautology

6 Ex. 34.5-6

Let HAM-CYCLE $= \{\langle G \rangle \mid G \text{ has a Hamiltonian cycle}\}$

HAM-PATH $= \{\langle G \rangle \mid G \text{ has a Hamiltonian path}\}$

We want to show that HAM-PATH is NPC.

First of all, HAM-PATH ∈ NP, since a certificate can be verified in $O(|V|)$ time.

We next show that HAM-CYCLE $\leq_{\mathrm{p}}$ HAM-PATH

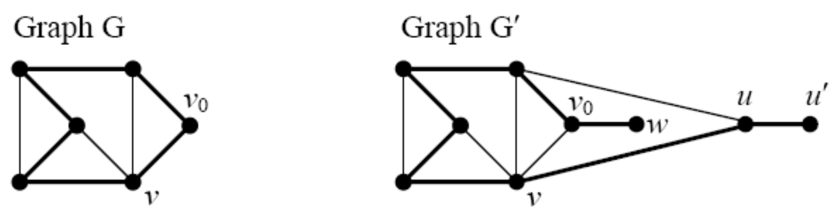
Given a graph $G = (V, E)$ for HAM-CYCLE, construct a graph $G' = (V', E')$ for HAM-PATH where

$V' = V \cup \{u, u', w\}$

$E' = E \cup \{(u, u'), (w, v_0)\} \cup \{(u, v) | (v, v_0) \in E\}$

where $v_0 \in V$ is an arbitrary vertex.

For example,



G has a Hamiltonian cycle $\Rightarrow$ G' has a Hamiltonian path

Let $(v_0, \dots, v, v_0)$ be a Hamiltonian cycle in G

Then, $(w, v_0, \dots, v, u, u')$ is a Hamiltonian path in G'

G' has a Hamiltonian path $\Rightarrow G$ has a Hamiltonian cycle

The Hamiltonian path in G' must be of the form $(w, v_0, \dots, v, u, u')$.

Since $(u, v) \in E'$, we have $(v, v_0) \in E$

Thus, $(v_0, \dots, v, v_0)$ is a Hamiltonian cycle in G.