CS581 Theory of Computation: Homework #6

Due on March 16 2016 at 12:30pm

Harry H. Porter Winter 2016

Konstantin Macarenco

Problem 6.1

Give an example in the spirit of the recursion theorem of a program in a real programming language (or a reasonable approximation thereof) that prints itself out.

Solution in python:

```
x = r"%sprint ('x = r\"' + x) %% (x + '\"\n')"
print ('x = r\"' + x) % (x + '\"\n')
```

Problem 6.11

Let ϕ_{eq} be defined as in Problem 6.10. Give a model of the sentence

$$\begin{aligned}
\phi_{lt} = \phi_{eq} & \quad (1)\\
\wedge \forall x, y[R_1(x, y) \rightarrow \neg R_2(x, y)] & \quad (2)\\
\wedge \forall x, y[\neg R_1(x, y) \rightarrow (R_2(x, y) \oplus R_2(x, z))] & \quad (3)\\
\wedge \forall x, y, z[(R_2(x, y) \wedge R_2(y, z)) \rightarrow R_2(x, z)] & \quad (4)\\
\wedge \forall x \exists y[_2(x, y)]. & \quad (5)
\end{aligned}$$

Solution

One model is $(N, R_1, R_2, \oplus)$, where R_1 is equality, R_2 is $<$ and $\oplus$ is $\vee$.

Problem 7.1

Answer each part TRUE or FALSE.

a.	$2n = O(n)$	**TRUE**
b.	$n^2 = O(n)$	**FALSE**
c.	$n^2 = O(n\ log^2 n)$	**FALSE**
d.	$n\ logn = O(n^2)$	**TRUE**
e.	$3^n = 2^{O(n)}$	**TRUE**
f.	$2^{2^n} = O(2^{2^n})$	**TRUE**

Problem 7.4
Fill out the table described in the polynomial time algorithm for context-free language recognition from theorem 7.16 for string $w = baba$ and CFG G:

$$\begin{aligned} S &\rightarrow RT \\ R &\rightarrow TR|a \\ T &\rightarrow TR|b \end{aligned}$$

Solution

	1	2	3	4
1	T	T,R	S	S,R,T
2		R	S	S
3			T	T,R
4				R

Because table(1,4) contains S, the TM accepts w.

Problem 7.5
Is the following formula satisfiable?

$$(x \vee y) \wedge (x \vee \overline{y}) \wedge (\overline{x} \vee y) \wedge (\overline{x} \vee \overline{y})$$

Solution

x	y	$(x \vee y)$	$(x \vee \overline{y})$	$(\overline{x} \vee y)$	$(\overline{x} \vee \overline{y})$	Result (conjunction of all)
T	T	T	T	T	F	F
T	F	T	T	F	T	F
F	T	T	F	T	T	F
F	F	F	T	T	T	F

Hence the formula is not satisfiable.

Problem 7.12
Call graphs G and H **isomorphic** if the nodes of G may be reordered so that it is identical to H. Let $ISO = \{\langle G, H\rangle | G \text{ and } H \text{ are isomorphic graphs}\}$. Show that $ISO \in NP$

Solution

A nondeterministic polynomial time algorithm for ISO operates as follows:
" On input $\langle G, H\rangle$ where G and H are undirected graphs:

1. Let m be the number of nodes of G and H. If they don't have the same number of nodes, $reject$.
2. Nondeterministically select a permutation π of m elements.
3. For each pair of nodes x and y of G check that (x,y) is an edge of G iff $(\pi(x), \pi(y))$ is an edge of H. If all agree, *accept*. If any differ, $reject$."

Stage 2 can be implemented in polynomial time nondeterministically, Stage 3 takes polynomial time. Therefore $ISO \in NP$