CS 381 Homework 6 solutions

October 26, 2000

Problem 15.1 $[\epsilon], [a], [aa], [b] \equiv_R [bx] \equiv_R [aax], [ab] \equiv_R [abx]$

	a	b
$\rightarrow [\epsilon]$	$[a]$	D
$[a]$F	$[aa]$	$[ab]$
$[aa]$F	D	D
$[ab]$F	D	$[ab]$
D	D	D

Problem 15.2

a) $[\epsilon], [01], [1], \forall k > 0\ [0^k]$ and $[0^{k+1}1]$

b) The relationship $\equiv_R$ is of infinite index.

c) $[\epsilon], \forall k > 0\ [0^k], [1^k]$.

Problem 16.1

$\{x \in a^* |$length is divisible by 2 or 7$\}$ and $\{x \in \{0,1\}^* | \#0(\text{x})$ is even and $\#1(\text{x})$ is divisible by 3$\}$

There are many correct answers. Here is one for each of the problems.

	$\vdash$	a	$\dashv$	comment
s	(s,R)	(q_0,R)	(t,L)	#a mod 2 = 0
q_0	-	(s,R)	(u,L)	#a mod 2 = 1
u	(p_0,R)	(u,L)	-	move to start
p_0	-	(p_1,R)	(t,L)	#a mod 7 = 0
p_1	-	(p_2,R)	(r,L)	#a mod 7 = 1
p_2	-	(p_3,R)	(r,L)	#a mod 7 = 2
p_3	-	(p_4,R)	(r,L)	#a mod 7 = 3
p_4	-	(p_5,R)	(r,L)	#a mod 7 = 4
p_5	-	(p_6,R)	(r,L)	#a mod 7 = 5
p_6	-	(p_0,R)	(r,L)	#a mod 7 = 6

	$\vdash$	0	1	$\dashv$	comment
s	(s,R)	(q_1,R)	(s,R)	(p_0,L)	#0 mod 2 = 0
q_1	-	(s,R)	(q_1,R)	(r,L)	#0 mod 2 = 1
p_0	(t,R)	(p_0,L)	(p_1,L)	-	#1 mod 3 = 0
p_1	(r,R)	(p_1,L)	(p_2,L)	-	#1 mod 3 = 1
p_2	(r,R)	(p_2,L)	(p_0,L)	-	#1 mod 3 = 2

Problem 17.1

a) not derivable because an a must follow every b.

b) $S \rightarrow AB \rightarrow AbA \rightarrow Aba \rightarrow aAba \rightarrow aaAba \rightarrow aaaAba \rightarrow aaaaba$

c) cannot derive bb because all b's must have an a immediately after them.

d) $S \rightarrow ABS \rightarrow ABAB \rightarrow ABAbA \rightarrow ABAba \rightarrow ABaba \rightarrow AbAaba \rightarrow Abaaba \rightarrow abaaba$

Problem 17.2

$$
\begin{aligned}
S &\rightarrow aAB | aBA | bAA | \epsilon \\
A &\rightarrow aS | bAAA \\
B &\rightarrow aABB | aBAB | aBBA | bS
\end{aligned}
$$

Prove that $L(G)$ is the language of all strings consisting of twice as many b's as a's. We will show this in two parts: first, we show that any string $x \in L(G)$ has twice as many a's as b's.

We wish to show inductively that the non-terminal S produces a string with $\#a(S) = 2\#b(S)$, and we will need to show that A produces a string with $\#a(A) = 2\#b(A) + 1$, and B produces a string with $\#a(B) = 2\#b(B) - 2$.

The induction here is on the structure of a derivation: for any string $x \in L(G)$, there is a derivation $S \to^* x$. At each step, the length of string produced by a non-terminal on the right side of a rule is smaller than the length produced by the non-terminal on the left. Thus our induction will terminate when we have a string of length 0 to produce (which is the base case).

Base cases:

- $S \to \epsilon$. Clearly the length of derivation here is one, and $\#a(\epsilon) = 2\#b(\epsilon) = 0$.
- $A \to aS \to a$. $\#a(a) = 1 = 2\#b(a) + 1$.
- $B \to bS \to b$. $\#a(b) = 0 = 2\#b(b) - 2$.

So we have base cases for each of the three non-terminals.

Inductive steps: We inductively assume that $\#a(S) = 2\#b(S)$, $\#a(A) = 2\#b(A) + 1$, and $\#a(B) = 2\#b(B) - 2$.

- $S' \to aAB$: We need to show that $\#a(S') = 2\#b(S')$, using inductive knowledge about A and B. So: $\#a(S') = 1 + \#a(A) + \#a(B) = 1 + [2\#b(A) + 1] + [2\#b(B) - 2] = 2[\#b(A) + \#b(B)] = 2\#b(S')$.
- $S' \to aBA$: same as above, by transitivity.
- $S' \to bAA$: $2\#b(S') = 2 + 2[2\#b(A)] = 2 + 2[\#a(A) - 1] = 2\#a(A) = \#a(S')$.
- $A' \to aS$: $\#a(A') = 1 + \#a(S) = 1 + 2\#b(S) = 1 + 2\#b(A')$.
- $A' \to bAAA$: $2\#b(A') = 2 + 3[2\#b(A)] = 2 + 3[\#a(A) - 1] = 3\#a(A) - 1 = \#a(A') - 1$.
- $B' \to aABB$: $\#a(B') = 1 + \#a(A) + 2\#a(B) = 1 + [2\#b(A) + 1] + 2[2\#b(B) - 2] = 2\#b(A) + 2[2\#b(B) + 1 + 1 - 4 = 2[\#b(A) + 2\#b(B)] - 2 = 2\#b(B') - 2$.
- $B' \to aBAB$: same as above by transitivity.
- $B' \to aBBA$: same as above by transitivity.
- $B' \to bS$: $2\#b(B') = 2 + 2\#b(S) = 2 + \#a(S) = 2 + \#a(B')$.

Part two is to show that any string x with $\#a(x) = 2\#b(x)$ can be derived by $S \to^* x$. We'll show this inductively with 3 simultaneous inductive hypotheses: if $\#a(x) = 2\#b(x)$ then $s \to^* x$, if $\#a(x) = 2\#b(x)+1$ then $A \to^* x$, and if $\#a(x) = 2\#b(x) - 2$ then $B \to^* x$.

Base cases: ϵ has $\#a(\epsilon) = 2\#b(\epsilon)$, and $S \to \epsilon$. a has $\#a(a) = 2\#b(a) + 1$ and $A \to aS \to a$. Similarly for b.

Inductive cases: Assume that for any string x with $|x| \leq n$ (where $n = 3k$ for some integer k) the S hypothesis holds, and with $|x| \leq n + 1$ the A and B hypotheses hold. Now we consider a string x with $|x| = n + 3$, and $\#a(x) = 2\#b(x)$. We have several cases to consider:

- $x = ayz$, where y and z are non-empty strings with $\#a(y) = 2\#b(y) + 1$ and $\#a(z) = 2\#b(z) - 2$. Then we use the production $S \to aAB$, and our inductive hypotheses give us $A \to^* y$ and $B \to^* z$.
- $x = azy$, where y and z have the same conditions as above. In this case, we use the production $S \to aBA$, and our inductive hypotheses work as before. (Note that since $\#a(y) + \#a(z) = 2\#b(y) + 2\#b(z) - 1$, so those are the only 2 cases with x beginning with a.)
- $x = byz$ with $\#a(y) = 2\#b(y) + 1$ and $\#a(z) = 2\#b(z) + 1$. In this case we have $S \to bAA$, and our inductive hypothesis gives us $A \to^* y$ and $A \to^* z$. (Note that this is the only case for x beginning with b because $\#a(y) + \#a(z) = 2\#b(y) + 2\#b(z) + 2$.)

We also must consider the cases for x, $|x| = n+4$ and either $\#a(x) = 2\#b(x)+1$ or $\#a(x) = 2\#b(x)-2$. These cases are similar to the above, with a few more possibilities for substrings, each corresponding to one rule in the grammar.