CS581 – Theory of Computation – HW7

Tuesday, May 21, 2013
due in class Tuesday, May 28, 2013

Answer each question below. You will turn this homework in using D2L. In addition, you may also turn in a paper copy in class. In this case the TA will mark up your homework with comments and return the comments to you.

You may format your answers using some document processing software, or you may write it up with pencil and paper and scan it. In either case submit a pdf document. Be sure your submission is clearly identified as Homework 7, and contains your name and your email on the first line. The first line should look like:

CS581 HW #7 Tom Smith tsmith@pdx.edu

1. **Infinite PDA is decidable.**
   let $\text{INFINITE}_{\text{PDA}} = \{< M > \mid \text{M is a PDA and L(M) is an infinite language}\}$.
   Show that $\text{INFINITE}_{\text{PDA}}$ is decidable.

2. **Proof by induction.**

   This question further extends the kinds of proofs we may do by induction. It introduces two new ideas. First, the inductively defined set may be constructed in more than 2 ways (in this example there are three ways to construct an Expression). Second, it introduces the idea that a proof (or a proof step) may have to be split into 2 or more cases. One of the induction steps will need to be split. This split arises because of the nature of the definition of function *distribute*. Note that it applies to only some *shapes* of Expressions.

   Consider the definition of Expressions, and some equations that describe two functions (value and distribute) over Expressions

   These are the 10 facts that you get to use.

```
1) If n is an integer, then (Const n) is an Expression.
2) It x and y are Expression, then (Plus x y) is an Expression.
3) It x and y are Expression, then (Times x y) is an Expression.
4) Nothing else is an Expression.
```

```
    An example Expression:  (Times (Const 3) (Plus (Const 1) (Const 6)))

5) value (Const n) = n
6) value (Plus x y) = (value x) + (value y)
7) value (Times x y) = (value x) * (value y)


8) distribute (Times x (Plus y z)) = Plus (Times x y) (Times x z)
9) distribute x = x      -- for every other shape of its argument

10) (x * a) + (x * b) = x * (a + b)
        A distributive law of addition over multiplication
```

Here is what you need to prove.

Prove by induction that `value(distribute x) = value x`.

Do the following in your solution.

- Write down what you are to prove.
- Make a formula (or function) parameterized by the induction variable.
- You will need to prove several cases. One for each way an Expression can be constructed. Express each case using your formula from above.
- A case might have 1 or more induction hypotheses. Use the formula as well to state what you can assume.
- Then work through the steps for each case.
- In one of the induction cases you will need to do a case analysis over one of the variables. Clearly state when you do so.

3. let $A = \{< R > \ | \text{R is a regular expression describing a language L}\}$. L contains at least one string that has 111 as a sub string (i.e. $w \in L$ and $w = x111y$, for some x and y). Show that A is decidable.

   Hint: Use the facts that regular languages are closed under complement and intersection, and that certain properties of DFA's are decidable.

4. Let $C_{CFG} = \{< G, k > \ | \text{ L(G) contains exactly } k \text{ strings}\}$. Where k is a finite constant, $k \geq 0$ or k is infinite. Show that C_{CFG} is decidable. Hint: How does a context free language accept an infinite set of strings?

5. **Reducability**. Let T = $\{< M > \ | \ M \text{ is a Turing Machine that accepts } w^R \text{ whenever it accepts } w\}$. Show that T is undecidable. Hint, this proof is very similar in structure to $\text{Regular}_{\text{TM}}$ from theorem 5.3, page 191. You might also look at the answer to question 5.28 in the text.