UNIVERSITY OF CALIFORNIA
Department of Electrical Engineering
and Computer Sciences
Computer Science Division

Prof. R. Fateman

Fall, 2001

Solutions: CS 164 Midterm 1: September 26, 2001, 9:00AM

1. [20 points] Here is a table describing an automaton with 2 states. The start state is S.

a. Draw a diagram of the automaton in the space to the right of the table. I'm not good at drawing in tex. The expected solution has two states, S and T. From S to itself is an arc labeled 0,1. From S to T is an arc labeled 0. From T to T is an arc labeled 0. T is a final stae, S has an incoming arc as the initial state.

b. Write down a simple regular expression that describes the same language that is recognized by this automaton.

`(0 | 1)*0` is the simplest solution.

c. In the space below, draw a DFA that accepts the same language. Use as few states as possible.

Two states, the same as above except that instead of an arc from S to S labeled 0,1 there is an arc from T to S labeled 1.

d. Write a context free grammar G0 that describes the same language.

Here's one.

```
X -> T0
T -> 0T | 1T | epsilon
```

2. [5 points] Write down a precise definition of $L(G)$ the language generated by any context free grammar G.

We expect something like

```
L(G) = {a1 a2 ...an | ai is in terminals(G), S==>* a1 a2 ... an, S is start(G)}
Or in English.. a set of all strings of terminal symbols derived from the start
symbol S using rules of G.
```

State	Transitions		Final State?
	0	1	
S	S,T	S	
T	T		yes

3. [6 points] Suppose grammar $G1$ has only one rule rewriting X, namely $X \rightarrow YZW$

a. If we know that $a \in$ First(Y), what can you conclude about First(X)?

A is in First(x)

b. Under what condition is First(W) $\subset$ First(X)?

if $\epsilon \in$ First(Y) and $\epsilon \in$ First(Z).

c. Under what condition is $\epsilon \in$ First(X)?

The condition above with $\epsilon \in$ First(W) also.

4. [5 points] Here are the rules for a grammar G2 with start symbol `S`

$$S \rightarrow aS$$

$$S \rightarrow b$$

Complete writing a recursive descent parsing program `parse` that returns `yes`, given a lisp list that constitutes a sentence in L(G2). We give you two useful parts already.

```
(defun parse (tokens)(s)(if (empty tokens) "yes"))

(defun eat(h) (cond((equal h (car tokens))(pop tokens))
                (t (error "stuck at ~s" tokens))))
;; sample test:  (parse '(a a b))

;; answer
(defun s()(case (car tokens)
           (a (eat 'a)(s))
           (b (eat 'b) t)))
```

5. [10 points] What is the result of running your Tiger lexical analysis program `fsl` on a file containing this material:

```
if then loop else23 >>>= 45
"hello /* world" iconst */
```

Run it to see the answer. It starts with ((if if (1 . 2)) (then then ...) ...)

6. [12 points]

On the next page is an LL(1) Parsing Table for a grammar G3 with start symbol E.

a. What are the rules of the grammar G3?

```
E -> TX
T -> iY | oEc
X-> pE | epsilon
Y-> mT | epsilon
\begin{verbatim}
```

	i	m	p	o	c	$
E	TX			TX		
X			pE		ϵ	ϵ
T	iY			oEc		
Y		mT	ϵ		ϵ	ϵ

```
b. What are the terminal symbols of G3?

i o c m p and maybe $

c. Trace each stack configuration in the parsing of the input
string {\tt o i c\$}. We have given you the first stack contents:
\begin{verbatim}
step  stack         input

1.       E $        o i c $
2.      TX$
3.      oEcX$
4.      EcX$
5.      TXcX$
6.      iYXcX$
7.      YXcX$
8.      XcX$
9.      cX$
10.      X$
11.      $

Note that this grammar, with the substitutions of m=*, p=+, o=( and c=)
should be familar to you.
```

7. [2 points]

a. Describe any unusual piece of clothing worn on Monday Sept. 24, by Prof. Fateman for the first 8 minutes of CS164 lecture.

You had to be there to see it.

b. How many CS164 lectures were delivered without the use of the video projector?

Ditto.