CS 536 Midterm Exam

Thursday, November 2, 1995

11:00 AM — 1:00 PM CST

Instructions

Answer question #1 and *any three* others. (If you answer all six, only the first four will count.) Point values are as indicated. Please try to make your answers neat and coherent. Remember, if we can't read it, it's wrong. Partial credit will be given, so try to put something down for each question (a blank answer always gets 0 points!).

Question #1

(1 point)
Halloween parties celebrate what late October holiday?

Question #2

(33 points)
Assume we add a simple parameterless macro facility to CSX. The command

```
#define ident  string
```

(in the file being scanned) will define identifier `ident` to be an abbreviation for `string`, an ordinary CSX quoted string. Whenever `ident` is scanned (as an identifier) after the `#define`, it will be replaced with the text of `string`. This text (with enclosing quotes removed and escaped characters expanded), is immediately scanned. `string` *may* contain macro calls.

Assume your scanner calls a subroutine `getNextChar()` whenever it wants the next character to be scanned. You may modify `getNextChar()` in any way you wish. Outline how you'd add this macro facility to your CSX scanner.

Question #3

(33 points)
Let #S# be the set of strings that contain an equal number of #a#'s and #b#'s (in any order). Thus #ab#, #abab#, #aababb# and #aabbaabbaabb# are all contained in #S#. Is #S# a regular set? If it is, give a regular expression or finite automaton that defines it. If #S# is not a regular set, explain carefully why no regular expression or finite automaton that defines #S# can be constructed.

Question #4

(33 points)
Let #s# be a string. Define #insert(s)# to be the function that inserts a ### into each possible position in #s#. If #s# is #n# characters long, #insert(s)# returns a set of #n+1# strings (since there are #n+1# places a ### may be inserted in a string of length #n#).

For example, #insert(abc)# = #{#abc,a#bc,ab#c,abc#}#. #insert# applied to a set of strings is the union of #insert# applied to members of the set. Thus #insert(ab,de)# = #{#ab,a#b,ab#,#de,d#e,de#}#.
Let #R# be any regular set (my choice). Show that #insert(R)# is a regular set.
Hint—Given a finite automaton for #R#, construct one for #insert(R)#.

Question #5

(a) (10 points)
Let Seq(x,y) be the set of all strings (of length 1 or more) composed of alternating x's and y's. For example, Seq(a,b) contains a, b, ab, ba, aba, bab, abab, baba, ... Write a regular expression that defines Seq(x,y).

(b) (23 points)
Let S be the set of all strings (of length 1 or more) composed of a's, b's, and c's, that start with an a and in which no two adjacent characters are equal. For example, R contains a, ab, abc, abca, acab, acac, ... but not c, aa, abb, abcc, aab, cac, ... Write a regular expression that defines S. You may use Seq(x,y) within your regular expression if you wish.

Question #6

(33 points)
Let #DFA# be any deterministic finite automaton (my choice). Assume you know #DFA# contains exactly #n# states and that it accepts at least one string of length #n# or greater. Show that #DFA# must also accept at least one string of length #2n# or greater.