

# CS 536

## Midterm Exam

Wednesday, March 29, 2006

11:00 AM — 1:00 PM

### Instructions

Answer question #1 and any three others. (If you answer more, 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!).

1. (1 point)

A scanner generator is used to generate what compiler component?

2. (33 points)

Let `NotEqual` be the set of strings that contain an unequal number of a's and b's (in any order).

That is,

`NotEqual = {a, b, aab, bba, abb, bab, baaa, bbba, ...}`.

Is `NotEqual` a regular set? If it is, give a regular expression or finite automaton that defines it.

If `NotEqual` isn't regular, explain carefully why.

3. (33 points)

Many languages, including C and C++, have a file insertion directive. In C and C++ it is of the form

```
#include filename
```

This command directs the compiler (and in particular the scanner) to read the contents of the named file as if it had been placed at the point where the `include` directive appeared. Note that nested includes are allowed (an included file may itself contain `#include` directives).

Explain how you would implement the `#include` directive in your CSX scanner.

What would happen if some file named `f` directly or indirectly included itself? How would you catch this error?

4. (33 points)

Let `SelectDigit` be the operator that selects strings that contain one or more digits (0 to 9) somewhere within them. For example, `SelectDigit({Abc, top10, 23, +10.0e22, two, p2c}) = {top10, 23, +10.0e22, p2c}`.

Let `R` be any regular expression (my choice). Show that `SelectDigit(R)` is a regular set.

5. (a) (13 points)

Is the set of strings that contain no duplicate characters regular? Why?

(b) (20 points)

Write a regular expression for comments that begin with << and end with >>. The body of the comment may contain any character sequence except >> (so that >> always marks the end of the comment).

6. Write JLex regular expression definitions that match the following strings

(a) (11 points)

The five characters: [`\n`]

(b) (11 points)

An identifier that begins with a letter, and is composed of letters, digits and zero or one underscore (e.g., `abc` or `a23` or `a_23`, etc.).

(c) (11 points)

A CSX quoted string that contains an even number of characters in its body. Thus `"ab"` and `"abcd"` and `"\n\n"` are allowed, but not `"a"` or `"\n"` or `"abc"`.