CS 536 Midterm Exam

Friday, March 24, 1995

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)
The C++ programming language was designed as a successor to what widely-used programming language?

Question #2

(33 points)
Let Double be the set of strings defined as {s|s=ww}. Double contains only strings composed of two identical repeated pieces. For example, if we have a vocabulary of the ten digits, 0 to 9, then the following strings (and many more!) are in Double: 11, 1212, 123123, 767767, 98769876, $\cdots$.

Assume we have a vocabulary consisting of the two letters, a and b. Is Double a regular set? Why?

Question #3

(33 points)
Let AllButLast be a function that returns all of a string but its last character. For example, AllButLast(abc) = ab. AllButLast(λ) is undefined. Let R be any regular expression that does not generate λ. AllButLast(R) is the set of strings denoted by R, with AllButLast applied to each string. Thus AllButLast(a^+b) = a^+. Show that AllButLast(R) is a regular set.

Question #4

(33 points)
Languages like C and C++ include *conditional compilation* facilities that allow parts of a program to be included or excluded from compilation.

Assume we add a conditional compilation facility to CSX. The symbol #if will be followed by an intlit. If the intlit is non-zero all the characters up to a #end symbol will be scanned as usual. The #if, the intlit and the #end will be deleted (just like white space or comments). If the intlit following the #if is zero, all the characters up to the #end will be skipped, as will the #if, the intlit and the #end. Thus if we have

```
#if   1
        a = b+c;
#end
```

the assignment will be scanned, but if we have

```
#if   0
        a = b+c;
#end
```

the assignment will be skipped by the scanner.

Outline how you'd add conditional compilation to you CSX scanner.

Should #if and #end pairs be allowed to nest? What must you do to support nested conditional compilation commands?

Question #5

(33 points)
Assume we add a new operation to our symbol table module:

```
void unRegister(serialNum sn);
```

UnRegister *removes* the registration of the string whose serial number is sn. That is, if symToStr(sn) == id and we unregister sn, then registerStr(id) is no longer guaranteed to be sn. UnRegister might be used for identifiers we know will no longer be used.

Explain how you'd implement unRegister, assuming your hash table uses chained resolution. What would you do if you used linear resolution?

After an identifier is unRegistered, should its serial number be reused? What are the advantages of reuse and what are the disadvantages?

Question #6

(33 points)
Assume we wish to write a program that examines a file that contains English language text (like a classnotes file) and lists the 10 words that appear most often. Outline how you might use Lex to facilitate writing this program. What would you do to make your Lex definition *ignore* common articles and connectives (like "the" and "or" and "duh")?