Midterm Review

PLT-4115
(Fall 2020)

Midterm Info

- Content:
  - Introduction to Compilers
  - Lexical Analysis

- Open-book and Open-notes.

- Students will have a 24-hour window
  - Wednesday, October 21 1:10 PM EST to Thursday, October 22 1:10 PM EST
  - The exam is timed (75 minutes) and need to take it in one sitting at some point during the 24-hour window.
  - Only one question can be answered at a time, and answers to previous questions are locked.
  - Previously-answered questions cannot be accessed again.
  - Questions and options are shuffled.
  - There will be only one attempt for the exam.

- It contains a mixture of short-answer and multiple-choice questions (number and weighting of questions are not disclosed to students).

- Solutions and students' responses will be released after everyone has taken the exam.
- Regrade requests can be made up to 1 week after the release of the grades.
- Clarification questions cannot be asked publicly on Piazza to prevent divulging information to students who haven't taken the exam yet.
- A test quiz will be available on Courseworks (containing simple fake questions);
  - the only purpose of this quiz is that students can get the "look and feel" of the exam they will take on Wednesday.
  - This test quiz will allow for multiple attempts, since it is only practice. Things to observe include the timed nature of the quiz, the restriction of only one quiz question appearing at a time (so that there are no surprises come exam time), the inability to access previous quiz questions, and the interface they will be interacting with for answering multiple-choice and short-answer questions. The test quiz will be available later tonight.

Intro: Q1

Q1. What advantages are there to a language-processing system in which the compiler produces assembly language rather than machine language?

Intro: Q2

Which of the following terms:

a. Imperative,

b. Functional

c. Scripting,

d. Static,

e. Dynamic

Apply to which of the following languages:

1. C

2. C++

3. Java

4. Scala

5. Perl

6. Python

7. JavaScript

Answer:
- C: imperative, static
- C++: imperative, static
- Java: imperative, static
- Scala: Functional, static
- Perl: Scripting, dynamic
- Python: Scripting, dynamic
- Javascript: Scripting, dynamic

Intro: Q3

What is printed by the following C code?

```c
#define a (x + 1)

 int x = 2;
 void b() { x = a; printf("%d\t", x); }
 void c() { int x = 1; printf("%d\n", a); }
 void main () { b(); c(); }
```
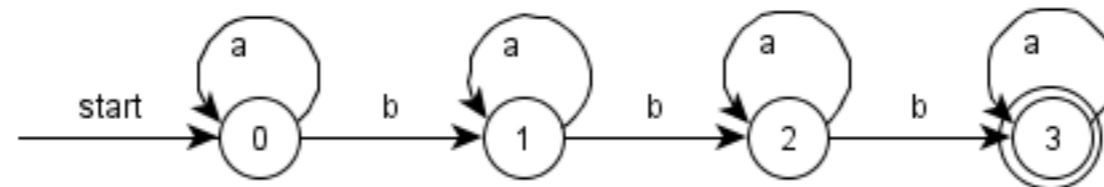
Answer: 3 2

Lexical Analysis: Q1

Try to describe the language defined by the following regular expressions:

1. `a(a|b)*a`
2. `a*ba*ba*ba*`
3. `(aa|bb)*((ab|ba)(aa|bb)*(ab|ba)(aa|bb)*)*`

Answer

1. A string of a and b that begins and ends with a.
2. A string of only 3 b consisting of a and b.
3. A string of a and b with an even number a and an even number b.

Lexical Analysis: Q2

Write the corresponding regular expression

Answer:

```
a* b a* b a* b a*
```

Lexical Analysis: Q3

Some languages, like SQL, are case insensitive, so a keyword can be written either in lowercase or in uppercase, or in any mixture of cases.
Example: he SQL keyword SELECT can also be written select, Select, or sElEcT, for instance.
Write a regular expression for the keyword "select" in a case insensitive language .

Answer: select -> [Ss][Ee][Ll][Ee][Cc][Tt]

Lexical Analysis: Q4

For the following C++ program:

```cpp
float capSqr(float r){
 return (r <= -10.0 || r >= 10.0) ? 100 :  r *r;
}
```

- Tokenize the program using appropriate lexemes and associated lexical values.
- Mention the lexical specifications (i.e., regular definition) you assumed to tokenize.

Lexical Analysis: Q5

- For an alphabet {0,1}, draw a DFA representing modulo 3 operations (The DFA only accepts strings divisible by 3 (e.g., 0,11,etc.))