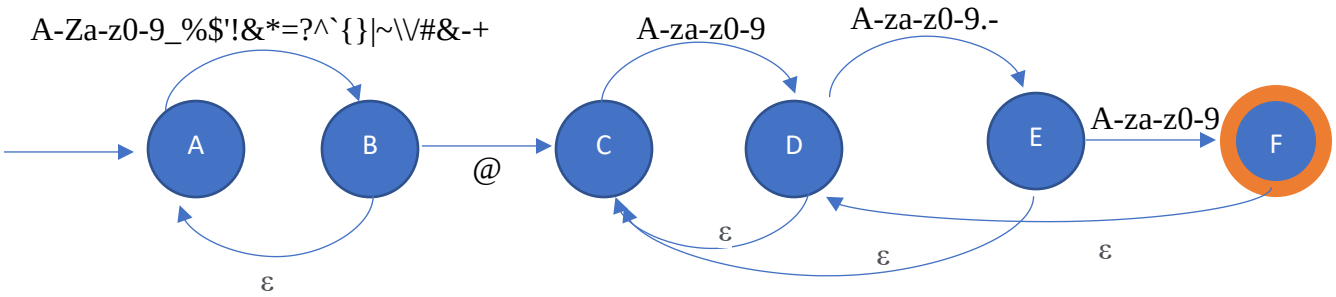
1. ^[^.](?!.*[.][.])[A-Za-z0-9_%$'!&*=?^`{}|~.\\/#&-+]*[^\\.]$

2. ^[^-.](?!.*[.][.])[A-za-z0-9-.]*[^\-.]$

3.



4. ^-?[0-9]*\.?([0-9]*|([0-9]*)?[eE]\-?[0-9])?[fFlL]?$

5. The four main evaluation criteria for programming languages are: readability, writability, reliability and cost. A floating-point constant is readable if it follows certain criteria. It can be negative, decimal, or whole number. It cannot start with zero unless that is the only character or the only character to the left of a decimal. If decimal there can only be one period and there must be numbers to the right of it. Exponent notation can be used with e or E, and the value will follow. Finally to declare type use f,F,l, or L. Ending without any of those characters will declare it type double. These criteria are implemented in the simplest way on regex to maximize writability and reliability.

6. Decimal Constant: ^[1-9][0-9]*(u|l)?$

Octal Constant: ^0[0-7]*(u|l)?$

Hexadecimal Constant: ^(0x|0X)[0-9A-Fa-f]*(I64|Ui64)?$

7. The four main evaluation criteria for programming languages are: readability, writability, reliability and cost. Integer literals have a lot of rules so I split them up into three parts for decimal, octal, and hexadecimal. This way the readability and writability will be improved rather than having one grammar for all three. Integer constants have high usage so they need to be streamlined. Writability is important for integer constants in order the distinguish between the different types and how they are declared.

11. ^(aa|bb|(ab|ba)(aa|bb)*(ba|ab))*(b|(ab|ba)(bb|aa)*a)[cd]*$

even number of a's and an odd number of b's followed by any number of c's or d's

12. Does not work because it does not include regex for a phrase to be followed by any number of c or d.

ex: aabbb, aabbbcd, aabbbcdc, aabbbcdcd

Also if the condition for even number and a's and odd number of b's is satisfied, and the phrase ends in b, then you cant add any more characters afterwards, because the last OR group in the regex are over written by the first two parts when those are satisfied.

ex. ababbaa, aabbbaa, aaaabbbaa, aabbaabbbaa

13. (aa|bb|aabb |abab| abba | baba |baab |bbaa)*b(aa|bb|aabb |abab| abba | baba|baab|bbaa)*|(aa|bb|aabb |abab| abba | baba |baab |bbaa)*b|b(aa|bb|aabb |abab| abba | baba |baab |bbaa)*

Move the last group to the front to fix regex.

17.

```
4 public class Question16{
11   public static void main(String[] args){
7   String str = "/* Line one of comment\nline two of comment\nline three of comment */";
5  System.out.println(str);
13   if (str.startsWith("/*") && str.endsWith("*/")){
7     System.out.println("True");
1   }
2   else{
7     System.out.println("False");
1   }
1   }
1}
```

Total: 60