Principles of Programming Languages [PLP-2014]
Exercises on Lexical Analysis - December 1, 2014

1. Write a regular expression that describes all integers x, such that x >15.

2. In words, describe the languages denoted by the following regular expressions:
   (a) 0*10*10*10*
   (b) (0 | 1) 1 (0|1)* 0 (0 | 1)

3. Depict the DFA that accepts the language that includes all strings of 0's and 1's with an even number of 0's and an odd number of 1's.

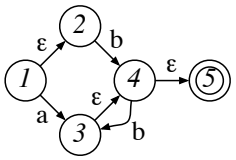
4. Consider the following Lex specification

```
digit          [0-9]
integer        {digit}({integer}|"")
%%
{integer}      { printf("integer: %s\n", yytext); }
%%
```

   Show what is wrong with the regular definitions in this specification. Fix the specification so that it correctly scans integer literals.

5. Use Thompson's algorithm to build an NFA for the regular expression ((a|ε)b)*

6. Given the NFA with S = {1,2,3,4,5}, Σ = {a,b}, s_0 = 1, F = {5} and the transition graph shown below, convert the NFA to a DFA using the subset construction algorithm (do not attempt to minimize the DFA). Express your answer as a transition graph and identify the start and final states.



7. Consider the following state transition table of a DFA with S = {0, 1, 2, 3, 4}, Σ = {a, b}, s_0 = 0, F = {3, 4}.

State	**a**	**b**
0	1	2
1	3	4
2	1	2
3	4	-
4	3	-

   (a) Draw the transition graph.
   (b) Minimize the DFA using the algorithm illustrated in class. Identify the start and final states of the minimized DFA.
   (c) Write an equivalent regular expression that represents the same language as defined by the (minimized) DFA.