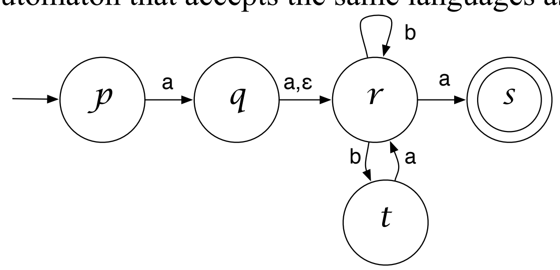
CS 365 – Spring 2020 – final problem set *submit solutions via Canvas by* Friday, May 1, 11:00pm

1. Let L_1 = `{x1y : x,y in {0,1}* and |x| + |y| is an even number}`.
   Show that L_1 is a regular language.

2. Let L_2 be the set of all strings over `{0,1}`* that have exactly one block of `0`'s of non-zero, even length. For this, "block" will mean all adjacent zeroes bounded on either side by an edge (left or right) of the string or a symbol `1`. Note that strings in L_2 may contain any number of odd-length blocks of `0`'s. Define a finite automaton to accept L_2.

3. Write a regular expression or regular grammar to generate exactly the set of binary representations of integers that are positive multiples of 3 – for example: **11** (3_{10}), **1100** (12_{10}), **101010** (42_{10}).

4. Construct a deterministic finite automaton that accepts the same languages as this ε-NFA (below)



5. Construct a minimal deterministic finite automaton that accepts $\{ 0^*11 \cup 01^*10 \}$

Let **min**(L) = { *w* | *w* is in L, but no proper prefix of *w* is in L)s
and **max**(L) = { *w* | *w* is in L, but there is no non-empty *x* such that *wx* is in L).

6. Show that regular languages are or are not closed under the **min** and **max** operations (above)

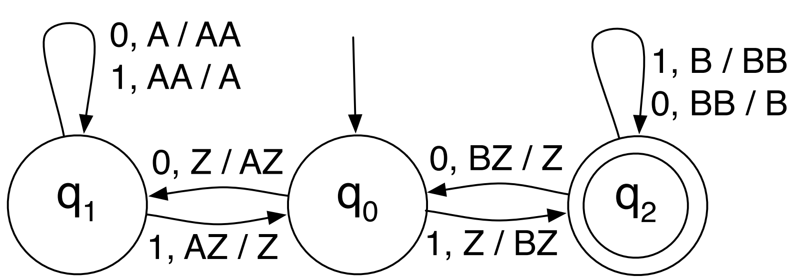
7. Show that context-free languages are or are not closed under the **min** and **max** operations (above).

8. Construct a pushdown automaton that accepts the language generated by this grammar:
   `S → 0S11 | 001`

9. Show that there is no deterministic pushdown automaton P such that L(P) = $\{0^n1^n \mid n \geq 1\} \cup \{0^n1^{2n} \mid n \geq 1\}$

10. Given a CFG G and one of its variables A, describe an algorithm to decide if there are any sentential forms in which A is the first symbol.

11. Represent the language of this pushdown automaton (below) as an equivalent context-free grammar.
    The language alphabet is {0,1}; the stack alphabet is {A,B} and Z represents an empty stack;
    Multiple stack values indicate top *n* stack values (*e.g.*, AZ means A on top of otherwise empty stack).



12. Show that this grammar is ambiguous or unambiguous:

```
S → A1B
A → 0A | ε
B → 0B | 1B | ε
```

13. Convert this grammar to Chomsky normal form:

```
S → AD | BSCD | C | ABD | ASE
A → 100A | E | F | 11 | ε
B → 11B | E11C | 1
C → 1C | 0011 | ε
D → 1D | 001 | 0
E → 1 | ε
F → 11E
```

14. Let $L_{14} = \{ xyz \mid \{x,y,z\} \in \{0,1\}^*, |x|>0, |y|>0, |z|>0, \#_0(y)=\#_1(y) \}$
This language contains all strings in form *xyz*, where *x*, *y*, *z* are each non-empty and the number of 0's in *y* is equal to the number of 1's in *y*.
Show that language L_{14} is regular, context-free (but not regular), or not context-free.

15. Let $L_{15} = \{ 0^x 1^y : x \text{ is even and } x \leq y \}$
Show that language L_{15} is regular, context-free (but not regular) or not context-free.

16. Let $L_{16} = \{ 0^a 1^b 2^c 3^d : a-c = b-d \}$
Show that language L_{16} is regular, context-free (but not regular) or not context-free.

17. Define a deterministic Turing machine that accepts an input string $w \in \{x,y,z\}^*$ and computes the number of x's in *w*. When the machine halts, the tape should contain only the unary representation of $f(w)$ followed by a zero. For example, f(yxxzyzxyy) = 1110.

18. Define a Turing machine that accepts an integer *n* as an input string in unary representation, $w \in \{1\}^*$, and calculates *n* mod 3. When the machine halts, the tape should contain only the unary representation of $f(w)$ followed by a zero. For example, f(1111) = 10 or f(111111111111) = 0.

19. Define a Turing machine that accepts an integer *n* as an input string in unary representation, $w \in \{1\}^*$, and calculates *n* / 3 (using integer division). When the machine halts, the tape should contain only the unary representation of $f(w)$ followed by a zero. For example, f(11111) = 10 or f(111111111111) = 11110.

20. Define a Turing machine that accepts $L_{20} = \{ xww^R y \mid x,w,y \in \{0,1\}^*, |w|>0, |x| \geq |y| \}$

21. Define an unrestricted grammar for $L_{21} = \{ 0^n w 1^n \mid n \geq 0, w \in \{0,1\}^*, |w|=n \}$

22. Define a Turing machine that accepts $L_{22} = \{ 0^n w 1^n \mid n \geq 0, w \in \{0,1\}^*, |w|=n \}$

23. Describe a Turing machine that accepts $\{ w_1\# w_2\#\ldots w_n \mid \forall 1 \leq k \leq n, w_k \text{ is the binary representation of } k \}$.
You do not need to list all states and transitions, just describe it clearly enough that its methodology is obviously feasible. You may employ any of the extensions to "normal" Turing machines that we discussed, such as nondeterminism, multiple heads or tapes, *etc*.

24. Describe a Turing machine M that takes as input a unary representation w of some integer n and leaves as output the unary representation of n^2 (with the tape head positioned at the leftmost cell of the output string). You may describe M by listing all tuples or by providing a sufficiently detailed textual description of the Turing machine's states and behaviors.

25. Note that we have never discussed ε-transitions in Turing machines (or linear bounded automata). What features of a Turing machine eliminate the necessity or value of ε-transitions, especially in light of their usefulness in finite automata and pushdown automata?

26. Imagine a Turing machine M' that works like a normal Turing machine except that whenever it moves to the left, it moves by two squares instead of one. Its movements to the right are by one square as usual. Show that M' can compute all functions that normal Turing machines can compute.

27. Given a Turing machine M and a string w, is it possible to determine whether M ever moves its tape head to the left when started with input w? If so, briefly explain how; if not, justify this problem as undecidable.

28. Show that this problem is undecidable: given a grammar G and a string w, does G generate w?

Identify each of the following statements as *true* or *false* and justify each response.

29. The union of infinitely many regular languages is regular.

30. If L is regular, then $L' = \{ xx : x \in L \}$ is also regular.

31. If L is regular, then $L' = \{ xy : x,y \in L \}$ is also regular.

32. All DFA-acceptable languages are context-free.

33. If there is a 10-state NFA that accepts some language L,
then there is a DFA with 50 states or less that accepts L.

34. If a problem is in NP, then no instance of that problem can be solved in polynomial time.

35. If $L_1 \leq_p L_2$ (L_1 can be reduced to L_2 in polynomial time) and $L_1 \in P$, then $L_2 \in P$.

36. Any context-free grammar in Chomsky normal form is unambiguous.