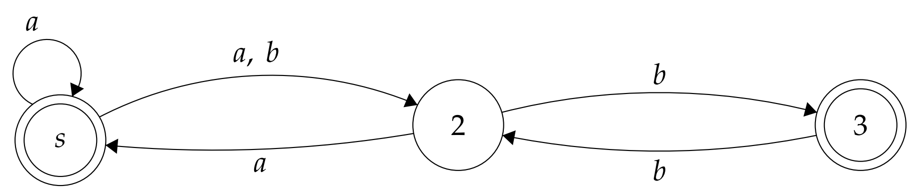
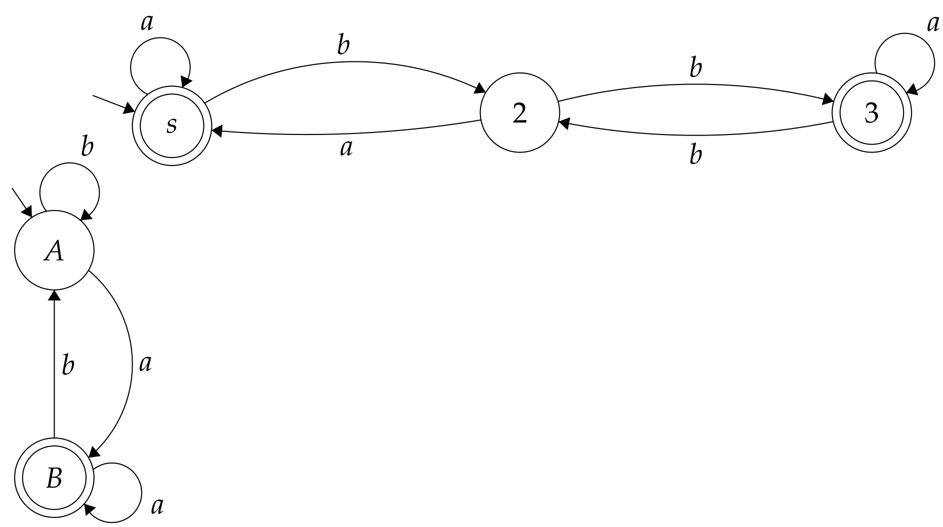
CSE396 **Problem Set 2** **Spring 2018**

Reading: Now read through the end of section 1.3—that is, read about "GNFAs" too. First focus on the proof in section 1.2 that every NFA has an equivalent DFA. Then before you read the proof of going from DFA/NFA/GNFA to a regular expression, reflect how it completes a cycle of showing that all four formalisms are equivalent in terms of the class of languages they denote: the *regular languages*. The GNFA-to-regexp proof may wrap over from Thursday into Tuesday of week 5. The notes `https://www.cse.buffalo.edu/~regan/cse396/CSE396.regexpalg` in the "Extra Resources" section of the course webpage give an alternate way to picture the algorithm in that proof.

(1) This problem is "HW2 Online Part" on *TopHat*, worth 20 pts. as before. Here is the NFA for the first five short-answer questions on it, which will help cut down scrolling:



(2) Call the following two DFAs M_1 (the one with three states that looks like N above) and M_2. Use the Cartesian product construction to design DFAs M_3 and M_4 such that $L(M_3) = L(M_1) \cap L(M_2)$ and $L(M_4) = L(M_1) \triangle L(M_2)$, where $\triangle$ is symmetric difference. (18 + 3 = 21 pts.)



(3) For each of the following languages A, write a regular expression r such that $L(r) = A$, and then give an NFA N_r such that $L(N_r) = A$. Well, if you give a DFA, that counts as an NFA, but in one or two cases you may find the NFA easier to build especially once you have r. For part (b), note that a string can be broken uniquely into maximal "blocks" of consecutive letters. For instance, in "Tennessee" the blocks are T, e, nn, e again, ss, and ee.

(a) The language of strings over $\{a, b\}$ in which every b is followed immediately by at least one a.

(b) The language of strings over $\{a, b\}$ in which every a belongs to a "block" of at least 2 a's and every b belongs to a block of at least 3 b's.

(c) The language of strings over $\{a, b\}$ with no block of 3 b's, and in which every block of 2 b's has an odd number of chars before it. (6 + 6 + 12 = 24 pts., for 65 total on the set)