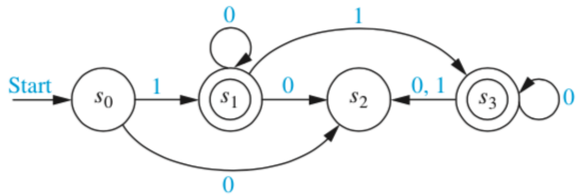
CSC 404 - Activity/Project 3 - Name:

Problem 1. Consider the following nondeterministic finite automaton (NFA) :



a. By drawing the 'tree of possibilities' (independent 'processes' or 'threads'), determine if the following bit strings/words are accepted or rejected by the machine.

$\mathtt{w}_1 = \mathtt{110}$ $\mathtt{w}_2 = \mathtt{101}$ $\mathtt{w}_3 = \mathtt{111}$ $\mathtt{w}_4 = \mathtt{100}$

b. Determine the language recognized by the NFA. You can write this out in 'words' or you can have a stab at writing it with fancy regular expressions (more about these soon!)

c. Construct a deterministic finite automaton (DFA) that recognizes the same language as the NFA.

Problem 2. In what follows, try to take advantage of nondeterminism as much as possible.

a. Construct a nondeterministic finite-state automaton that recognizes the set of all bit strings that contain three `0`s.

b. Construct a nondeterministic finite-state automaton that recognizes the set of all bit strings of at least 4 characters that begin and end with `11`.

c. Construct a nondeterministic finite-state automaton that recognizes the set of all bit strings that contain the substring `0101` (i.e., $\texttt{w} = \texttt{x0101y}$ for some `x` and `y`). Use at most 5 states.

d. Construct a nondeterministic finite-state automaton that recognizes the set of all bit strings that contain an even number of `0`s or contain exactly two `1`s.

e. Construct a nondeterministic finite-state automaton that recognizes the set of strings over the alphabet $\{a, b, c, d\}$ such that the final character has appeared before. For example,

`abca, bcdaa, dad, bbb` are accepted and `abc, cba, adc` are rejected.