# TOPIC ONE: Non deterministic Machines

## Question 1.

Ex. 2.17. Consider the problem of taking an arbitrary nondeterministic finite-state machine that accepts some language $A$ and modifying the machine to accept the complement of $A$ instead (i.e., all the strings that are not in $A$).

Does the following approach work: Make all rejecting states of the nondeterministic machine into accepting states and all accepting states into rejecting states?

Briefly explain your answer.

## Question 2.

Ex. 2.13. For any $k \geq 0$ let $L_k$ be the language of bitstrings whose first $k$ bits equal their last $k$ bits. How many states does a deterministic finite-state machine for $L_k$ need? Same for a nondeterministic machine. How long is a regular expression for $L_k$?

## Question 3.

Ex. 2.19. Give three properties that are decidable for finite-state machines but undecidable for arbitrary programs.

# TOPIC TWO: Cross- Product Construction

## Question 1.

Ex. 2.21. Consider creating the cross-product machine of two given machines and choosing the accepting states such that the new machine accepts the intersection of the languages of the two old machines.

*(a)* Give an example where the cross-product machine is minimal.

*(b)* If the cross-product machine is minimal, does that imply that the two "factors" in the product had to be minimal? Give an argument or a counterexample.

## Question 2.

Ex. 2.22. For any program $p$ define $L_p$ to be the set of input strings on which the program prints the word "Yes" and terminates. Is it decidable whether or not a program $p$ has a regular language $L_p$? Briefly justify your answer.

## Question 3.

Ex. 2.24. Construct a finite-state machine for the language $\{x : x \in \{0,1\}^\star$ *and* $x \notin \{000, 001\}^\star\}$. First construct a nondeterministic machine that is as small as possible, then apply the subset construction, then minimization.

## Question 4.

Ex. 2.28. Formulate a stronger version of the pumping lemma, one that says not only that pumping occurs somewhere in a long enough string but that it occurs within every long enough substring. Use this stronger version to prove two languages from [8] Figure **??** not regular which cannot be proven not regular with the original weaker version.