

Homework 4

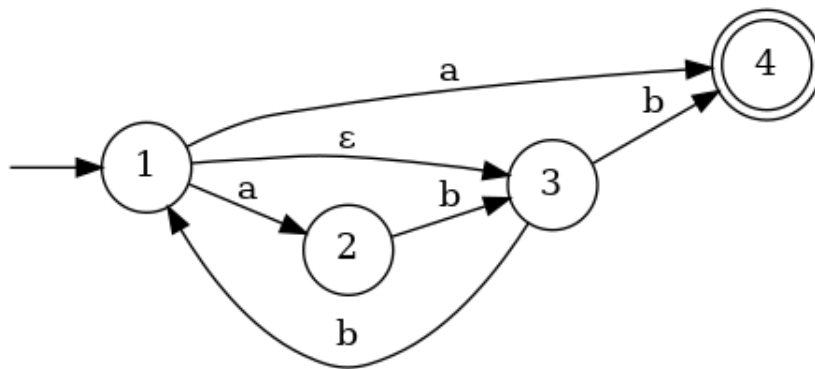
CptS 317, Spring 2021

Due Date: March 10th, 2021 by 11:59pm Pacific.

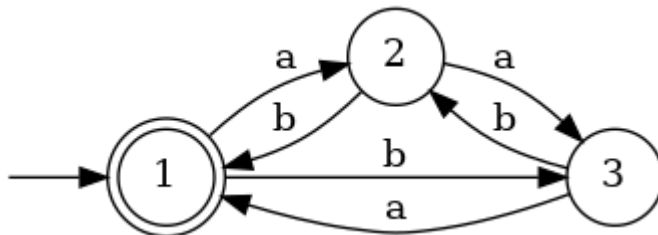
To be submitted on Canvas.

This homework has six problems, each of which is equally weighted.

1. Convert the following NFA into a DFA:



2. Convert the following DFA into a 2-state GNFA using the iterative process discussed in class and in the textbook (i.e. first turn into a 5-state GNFA, then remove one state at a time)



3. Describe the error in the following “proof” that 0^*1^* is not a regular language (an error must exist because 0^*1^* is regular). The proof is by contradiction.

- Assume 0^*1^* is regular.
- Let p be the pumping length for 0^*1^* given by the pumping lemma.
- Choose s to be the string 0^p1^p .
- We know that s is a member of 0^*1^* , but a previous proof[†] shows that s cannot be pumped.
- Thus there is a contradiction. So 0^*1^* is not regular.

[†]*This is demonstrated both in example 1.73 in your textbook and in example 1 of the Pumping Lemma lecture slides on Canvas.*

4. Let $\Sigma = \{1, \#\}$ and let $Y = \{w \mid w = x_1\#x_2\#\cdots\#x_k \text{ for } k \geq 0, \text{ and each } x_i \in 1^*, \text{ and } x_i \neq x_j \text{ for } i \neq j\}$

Prove that Y is not regular using the pumping lemma.

5. Give context-free grammars generating the following languages. In all parts, the alphabet $\Sigma = \{0, 1\}$

- a) $\{w \mid w \text{ starts and ends with the same symbol}\}$.
- b) $\{w \mid \text{the length of } w \text{ is odd}\}$.
- c) $\{w \mid w = w^R, \text{ that is, } w \text{ is a palindrome}\}$.
- d) The empty set.

6. The following is a description for “JH-Lisp2”, which is a **modified** version of the language previously described in homework 3, problem 6:

- (a) The alphabet of JH-Lisp2 consists of the open '[' and close ']' square brackets, the digits 0–9, the underscore character '_', and the letters {a,b,d,i,l,m,o,s,u,v}
- (b) An **operator** is a substring in JH-Lisp2 containing one of the following strings: {add, sub, mul, div, mod}.
- (c) A **number** is a substring in JH-Lisp2 containing one or more of the digits 0–9
- (d) An **expression** is defined as either a **number** or a **list expression**
- (e) A **list expression** is defined by the following, in order:
 - i. An open bracket: '['
 - ii. An **operator**
 - iii. An underscore: '_'
 - iv. An **expression**
 - v. An underscore: '_'
 - vi. An **expression**
 - vii. A close bracket: ']'
- (f) A string in the JH-Lisp2 language is a single **list expression**.

An example string in this language might be [add_[mod_4_3]_21]

Give a context-free grammar for JH-Lisp2.