1.65 Prove that for each \( n > 0 \), a language \( B_n \) exists where
a. \( B_n \) is recognizable by an NFA that has \( n \) states, and
b. if \( B_n = A_1 \cup \cdots \cup A_k \) for regular languages \( A_i \), then at least one of the \( A_i \) requires a DFA with exponentially many states.

1.66 A homomorphism is a function \( f: \Sigma \rightarrow \Gamma^* \) from one alphabet to strings over another alphabet. We can extend \( f \) to operate on strings by defining \( f(w) = f(w_1)f(w_2)\cdots f(w_n) \), where \( w = w_1w_2\cdots w_n \) and each \( w_i \in \Sigma \). We further extend \( f \) to operate on languages by defining \( f(A) = \{ f(w) \mid w \in A \} \), for any language \( A \).

a. Show, by giving a formal construction, that the class of regular languages is closed under homomorphism. In other words, given a DFA \( M \) that recognizes \( B \) and a homomorphism \( f \), construct a finite automaton \( M' \) that recognizes \( f(B) \). Consider the machine \( M' \) that you constructed. Is it a DFA in every case?
b. Show, by giving an example, that the class of non-regular languages is not closed under homomorphism.

1.67 Let the rotational closure of language \( A \) be \( RC(A) = \{ yx \mid xy \in A \} \).

a. Show that for any language \( A \), we have \( RC(A) = RC(RC(A)) \).
b. Show that the class of regular languages is closed under rotational closure.

1.68 In the traditional method for cutting a deck of playing cards, the deck is arbitrarily split two parts, which are exchanged before reassembling the deck. In a more complex cut, called Scanne's cut, the deck is broken into three parts and the middle part in placed first in the reassembly. We'll take Scanne's cut as the inspiration for an operation on languages. For a language \( A \), let \( CUT(A) = \{ yxz \mid xyz \in A \} \).

a. Exhibit a language \( B \) for which \( CUT(B) \neq CUT(CUT(B)) \).
b. Show that the class of regular languages is closed under \( CUT \).

1.69 Let \( \Sigma = \{0,1\} \). Let \( WW_k = \{ wu \mid u \in \Sigma^* \} \) and \( w \) is of length \( k \).

a. Show that for each \( k \), no DFA can recognize \( WW_k \) with fewer than \( 2^k \) states.
b. Describe a much smaller NFA for \( \overline{WW_k} \), the complement of \( WW_k \).

1.70 We define the avoids operation for languages \( A \) and \( B \) to be \( A \) avoids \( B = \{ w \mid w \in A \text{ and } w \text{ doesn't contain any string in } B \text{ as a substring} \} \).

Prove that the class of regular languages is closed under the avoids operation.

1.71 Let \( \Sigma = \{0,1\} \).

a. Let \( A = \{ \alpha^i \alpha^j \mid k \geq 1 \text{ and } u \in \Sigma^* \} \). Show that \( A \) is regular.
b. Let \( B = \{ \alpha^i \beta^j \mid k \geq 1 \text{ and } u \in \Sigma^* \} \). Show that \( B \) is not regular.

1.72 Let \( M_1 \) and \( M_2 \) be DFAs that have \( k_1 \) and \( k_2 \) states, respectively, and then let \( U = L(M_1) \cup L(M_2) \).

a. Show that if \( U \neq \emptyset \), then \( U \) contains some string \( s \), where \( |s| < \max(k_1, k_2) \).
b. Show that if \( U \neq \Sigma^* \), then \( U \) excludes some string \( s \), where \( |s| < k_1k_2 \).

1.73 Let \( \Sigma = \{0,1,\#\} \). Let \( C = \{ x\#x\#x \mid x \in \{0,1\}^* \} \). Show that \( C \) is a CFL.