



# **Tutorial Letter 203/1/2018**

## **Theoretical Computer Science 2**

### **COS2601**

**Semester 1**

**School of Computing**

Discussion of Assignment 3

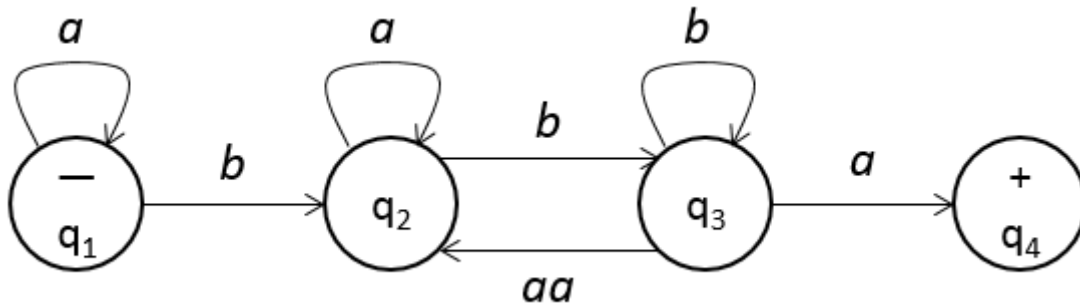
Dear student,

Solutions to the questions of assignment 03 are provided in this tutorial letter.

Regards,  
COS2601 team

**Question 1**

Kleene's theorem can be used to turn a transition graph (TG) into a regular expression. Which one of the following regular expressions would generate a language that would be equivalent to the language described by the following TG?

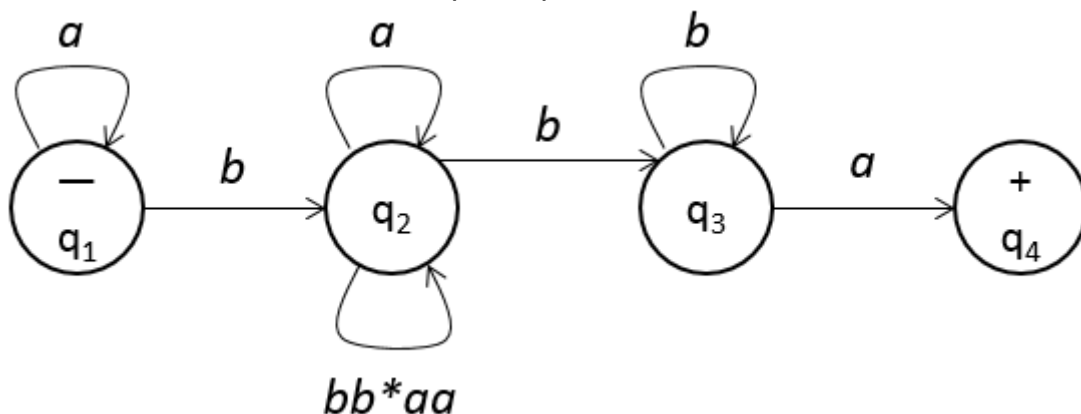


1.  $a^*b(a + bb^*aa)^*(b + aaa^*b)^*a$
2.  $a^*b(abb^*aa)^*bb^*a$
3.  $a^* + b + (a + bb^*aa)^* + b + (b + aaa^*b)^* + a$
4.  $a^*b(a + bb^*aa)^*bb^*a$

Answer: Option 4

*Discussion*

The  $a$ -loop at  $q_1$  can be written as  $a^*$ . Similarly the other 2 loops at  $q_2$  and  $q_3$  to get  $a^*$  and  $b^*$ . The loop from  $q_2$  to  $q_3$  and back to  $q_2$  can be written as a loop at  $q_2$  as  $(bb^*aa)$ : follow the  $b$  edge to  $q_3$ , loop on  $b$  at  $q_3$ , and return along the  $aa$  edge to  $q_2$ . We can then remove the  $aa$  edge as it is now included in the loop at  $q_2$ . Note that we do this as a single step so as not to get confused. Only then do we consider how to write the 2 loops at  $q_2$ .



The 2 loops at  $q_2$  would then be written as  $(a + bb^*aa)^*$  - that is, we can follow either one, or both, or neither, of the 2 loops.

It is now simply a matter of concatenating these parts of the regular expression to get to

$$a^*b(a + bb^*aa)^*bb^*a$$

From this, it is clear that option 4 is the correct option.

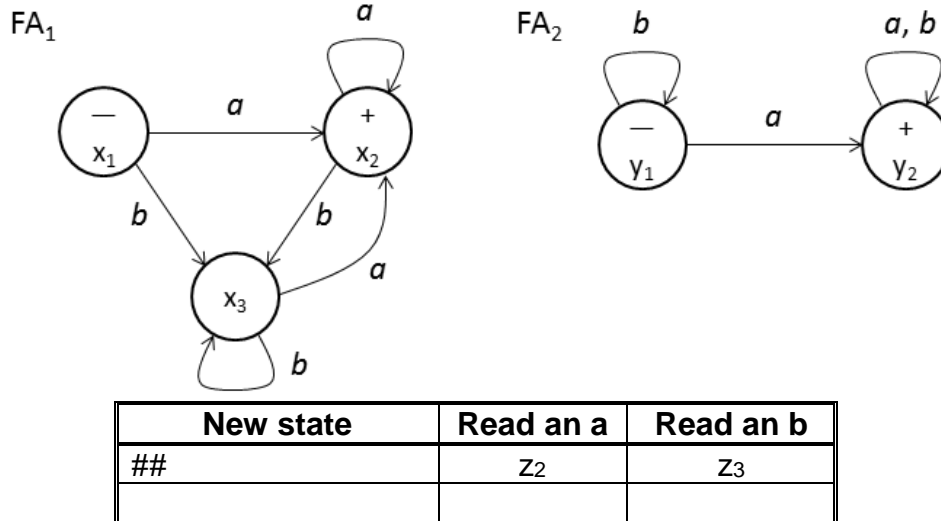
Option 1,  $a^*b(a + bb^*aa)^*(b + aaa^*b)^*a$ , is not correct as it misses the  $b$  edge between  $q_2$  and  $q_3$ .  
Option 3,  $a^* + b + (a + bb^*aa)^* + b + (b + aaa^*b)^* + a$ , is not correct as it misinterprets the

concatenations from one state to the next as options, implying that one chooses either the first part,  $a^*$ , or the second part,  $b$ , or the loop, and so on, when generating words, and this would generate words not accepted by the FA, for example,  $b$ .

Option 2,  $a^*b(abb^*aa)^*bb^*a$ , is not correct as it concatenates the 2 loops at  $q_2$  rather than interpreting them as optional paths.

**Question 2**

Given  $FA_1$  (with regular expression  $r_1$ ) and  $FA_2$  (with regular expression  $r_2$ ), a transition table is being put together to build an FA for  $r_1 + r_2$ .



Given the partial transition table above, which one of the following would be entered into the cell indicated by ## (the first cell in the second row)?

1.  $-Z_1 = x_1$
2.  $-Z_1 = x_1$  or  $y_1$
3.  $\pm Z_1 = x_1$  or  $y_1$
4.  $\pm Z_1 = x_1$

Answer: Option 2

*Discussion*

As we are looking for the union of  $FA_1$  and  $FA_2$  (that is, words that are accepted by either one of the machines), we need to start our new FA in the initial states of both  $FA_1$  and  $FA_2$  at the same time: giving  $x_1$  or  $y_1$ . Also, this will be an initial state only (as neither  $x_1$  nor  $y_2$  are final states), giving  $-z_1$ . Thus option 2 would be the correct option.

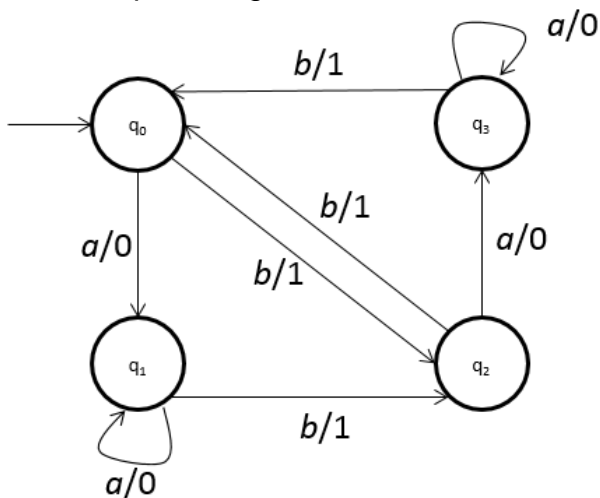
Option 1 is incorrect as the new machine is only starting on  $FA_1$ .

Option 3 is incorrect as it makes the initial a final state as well, which it is not.

Option 4 is the initial state for the closure of  $FA_1$ , that is  $FA_1^*$ .

**Question 3**

Given the Mealy machine below, which one of the following represents the output that would be printed if the machine were fed the input string *baabb*?



1. 010011
2. 10011
3. 110011
4. None of the above

Answer: Option 2

*Discussion*

We trace the operation of this machine on the input *baabb*. For a Mealy machine, we start in  $q_0$ . Then we read the first letter of the input: *b*. This takes us to state  $q_2$ , and a 1 is printed. Continue to follow the trace in the table below.

Read	Move to state	Print
Start	In $q_0$	nothing
<i>b</i>	$q_2$	1
<i>a</i>	$q_3$	0
<i>a</i>	$q_3$	0
<i>b</i>	$q_0$	1
<i>b</i>	$q_2$	1

Reading from the table, 10011 is printed, and thus option 2 is the correct option.

**Question 4**

Which one of the following is a Mealy machine that recognises the sequence 0110? When such a sequence is recognised anywhere in an input string, the output should be 0001; that is, output 0 until the sequence 0110 is discovered, when a 1 should be output. Also, 0110110 contains two 1001 sequences, even though there is some overlap.

- 1.
- 2.
- 3.
- 4.

Answer: Option 1

*Discussion*

Let's consider a string that contains more than one 0110 substring, and check whether the options provide the required output. If a machine fails to recognise any of the 0110 substrings (including where there is an overlap of the first and last 0), then it cannot be the correct answer. Thus, 0110110 should produce the output 0001001.

Option	0110110
1	0001000
2	0001000
3	0001111
4	0001001

Clearly, only option 1 correctly identifies the 1001 sequences correctly, and should be the selected option.

### Question 5

Which one of the following statements about regular languages is true?

1. If  $L$  is a regular language, then  $L'$  is also a regular language.
2. If  $L_1$  and  $L_2$  are regular languages, then  $L_2L_1$  is not necessarily a regular language.
3. If  $L_1$  and  $L_2$  are regular languages, then  $L_1 \cap L_2$  is not necessarily a regular language.
4.  $L_1 \cap L_2 = L_1' \cup L_2'$

Answer: Option 1

#### *Discussion*

##### *Option 1*

See Theorem 11 on page 173 of Cohen – regular languages are closed under complementation. This statement is true, and is thus the correction option to choose.

##### *Option 2*

See Theorem 10 on page 169 of Cohen – regular languages are closed under concatenation; the order of the concatenation does not make a difference. This statement is false.

##### *Option 3*

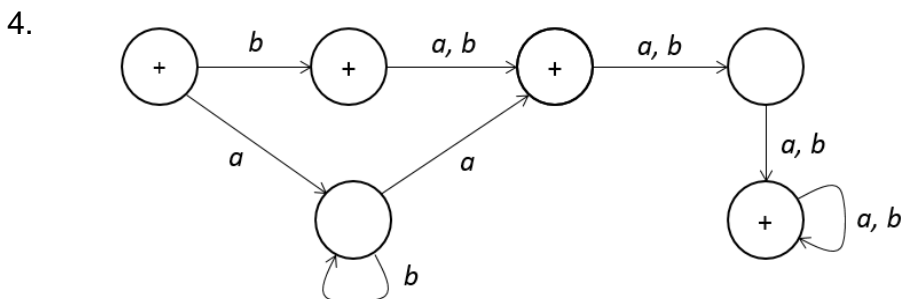
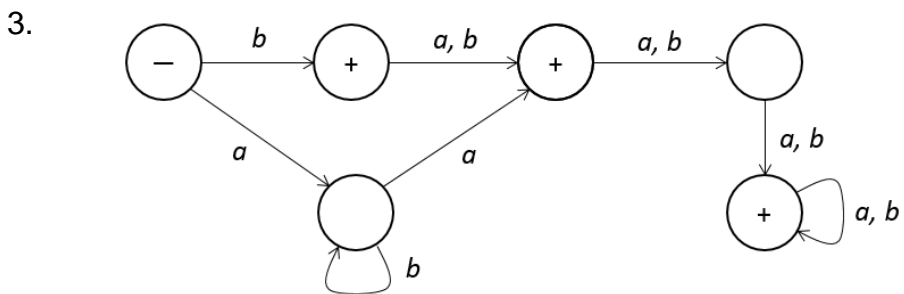
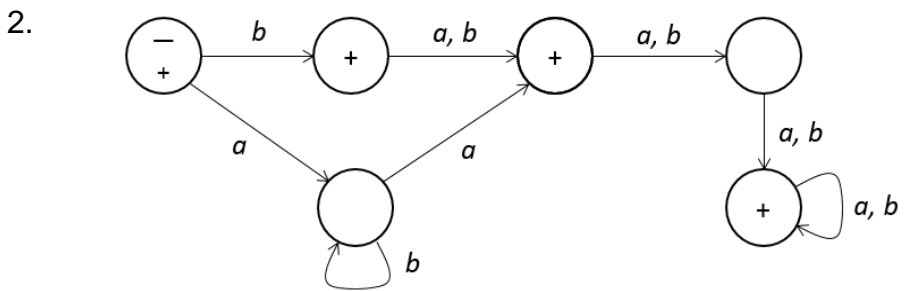
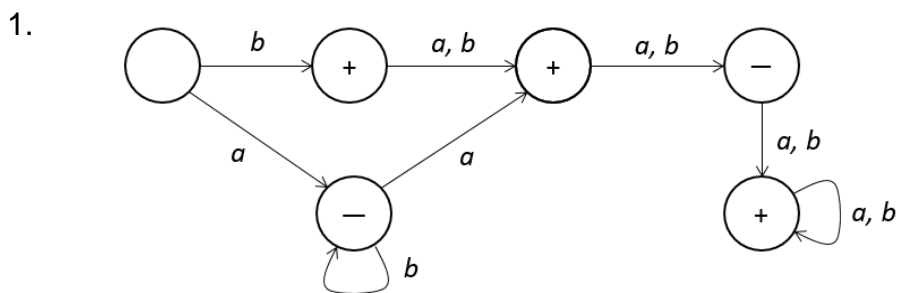
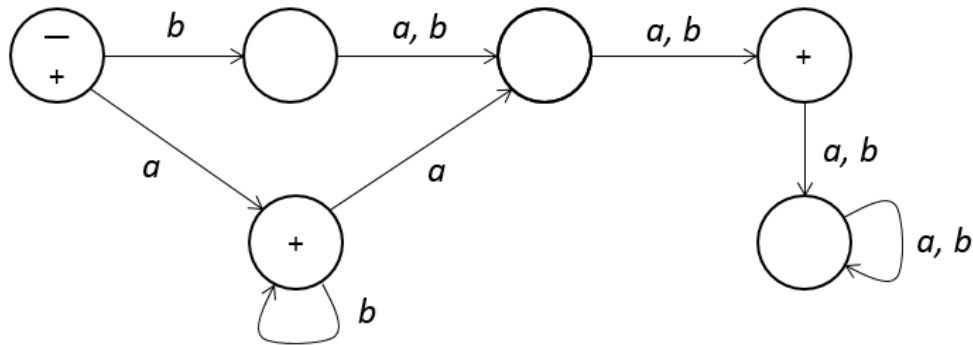
See Theorem 12 on page 174 of Cohen – regular languages are closed under intersection. This statement is false.

##### *Option 4*

See the start of the proof of Theorem 12 on page 174 of Cohen – De Morgan's Law states that  $L_1 \cap L_2 = (L_1' \cup L_2)'$  or  $(L_1' \cup L_2)'$  – it is the complement of the union of  $L_1'$  and  $L_2$ . This statement is false. As an aside, de Morgan's Law is true for all sets, whether regular languages or not.

**Question 6**

Given the FA (representing the language L) below, which one of the following is the FA that represents the language L'?



Answer: Option 3

*Discussion*

To create the complement of an FA, the following needs to be done.

1. All final states are made non-final states.
2. All non-final states are made final states.
3. Note that if the start state is a non-final state (–) it comes a final state too (±), and if the start state is a final state (±) it then becomes a non-final start state (–).

*Option 1*

This is not the correct option as the final states were changed to start states instead of being changed to non-final states.

*Option 2*

This is quite close to the correct solution, but it is wrong because the final start state was not made a non-final state.

*Option 3*

This is the correct option as

1. the final start state was made a non-final state,
2. all final states were made non-final states, and
3. all non-final states were made final states.

*Option 4*

Again, this is quite close to the correct solution, but it is wrong because the final start state was made a final start state instead of a non-final start state.

**Question 7**

---

When using the pumping lemma with length to prove that the language  $L = \{a^2b^nab, n>0\}$  is nonregular, the following approach is taken.

Assume L is regular.

Then there exists an FA with k states which accepts L.

We choose any word w which is a word in L.

Which one of the following would be an appropriate word w?

1.  $a^2b^2ab$
2.  $a^2b^mab$  for  $m>0$
3.  $a^2b^ka^kb^k$
4.  $a^2b^kab$

Answer: Option 4

*Discussion*

To use the pumping lemma with length, we need to choose a word w such that it has more letters than the number of states in the FA that accepts the language, which is k in this case.

*Option 1*

We cannot choose this option as we cannot be certain that  $\text{length}(a^2b^2ab)$  is greater than k, which we need when using the pumping lemma with length.

*Option 2*

We cannot choose this option for the same reason that we could not choose option 1 – we cannot be sure that  $\text{length}(w)$  is greater than  $k$ , which we need to know when using the pumping lemma with length.

*Option 3*

We cannot choose this option, even though its length is greater than  $k$ , as  $a^2b^ka^kb^k$  is not a word in  $L$  – it has  $(n-1)$  more  $a$ 's and  $b$ 's at the end of the word than it should, and thus does not match the pattern for words in  $L$ .

*Option 4*

This is the correct option as we need a word  $w$  where  $\text{length}(w) > k$ , and in this case,  $w$  has  $(k+4)$  letters.

**Question 8**

Consider the following statements.

- A. There exists an FA that accepts the nonregular language  $\{a^n b^{n+1} \text{ where } n \geq 1\}$ .
- B. The nonregular language  $\{a^n b^n \text{ where } n \geq 0\}$  can be written as the regular expression  $\mathbf{a^*b^*}$ .
- C. The language accepted by an FA can be a nonregular language.
- D. The *reductio ad absurdum* approach can be used to prove that a language is not regular.

Which one of the following correctly identifies true statements about nonregular languages?

- 1. All the statements are true.
- 2. Only D is true.
- 3. Only A, B, and C are true.
- 4. None of the statements is true.

Answer: Option 2

*Discussion*

We first consider the 4 options.

*Statement A*

By the definition of a nonregular language, such a language cannot be defined by a regular expression, and thus by Kleene's Theorem, there can also not be an FA that accepts it. The given statement is thus not true.

*Statement B*

The given statement is false as nonregular languages cannot, by definition, be defined by a regular expression.

*Statement C*

The given statement is false as a language that can be accepted by an FA can be defined by a regular expression (by Kleene's Theorem), and by definition a language that can be defined by a regular expression is called regular.

*Statement D*

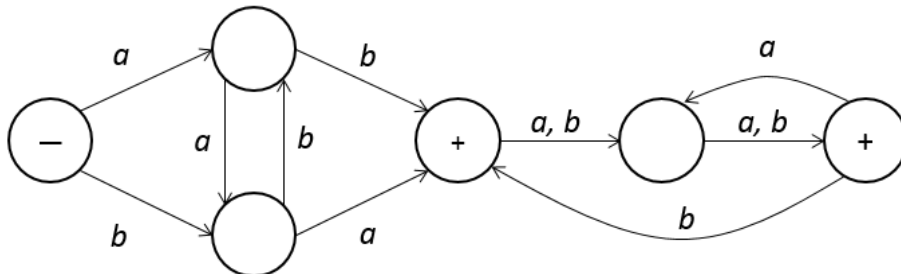
The given statement is true as it is possible to begin by assuming a language is regular, and then showing how this assumption leads to a contradiction, thus leading to the rejection of the

assumption. Cohen uses this technique on page 193 to prove that  $a^n b^n$  is nonregular. Thus this is the option that should be chosen.

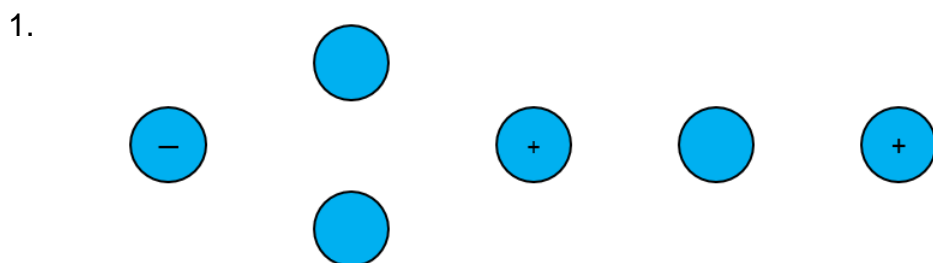
As only statement D is true, option 2 is the correct answer.

**Question 9**

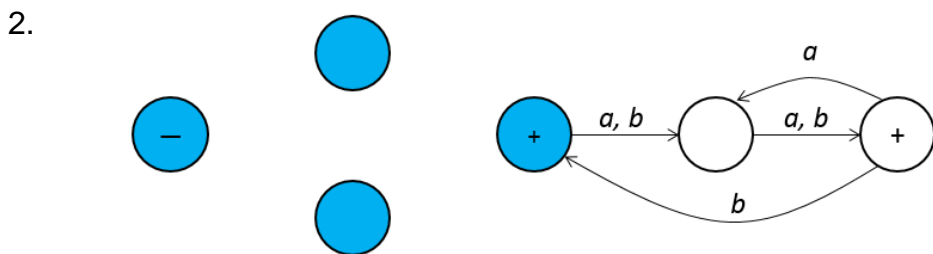
Use the blue paint procedure to determine whether the following FA accepts any words.



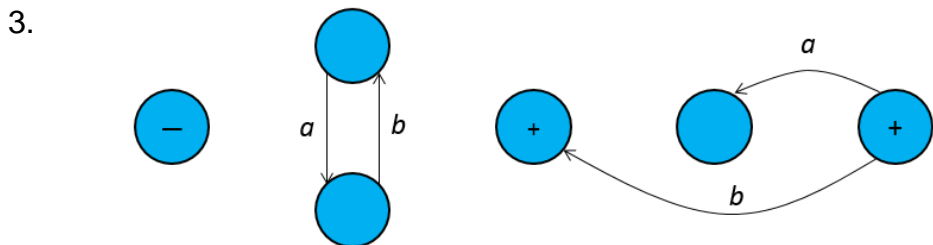
Which one of the following represents the correct final step in the procedure with the correct conclusion?



From this we can determine that the machine accepts some words.

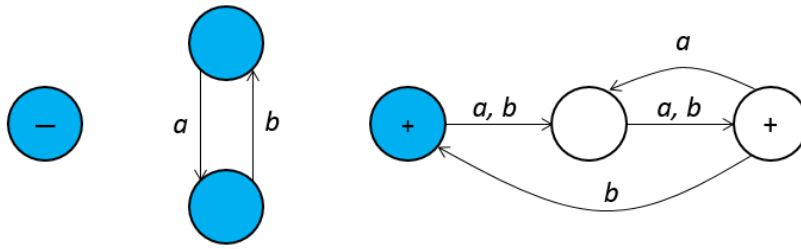


From this we can determine that the machine accepts some words.



From this we can determine that the machine accepts some words.

4.



From this we can determine that the machine accepts some words.

Answer: Option 1

*Discussion*

Following the blue paint algorithm on page 210 will give the correct solution.

1. Paint the start state blue.
2. From every blue state, follow each edge that leads out of it and paint the destination state blue, and then delete the edge from the machine.
3. Repeat step 2 until no new state is painted blue.
4. If **any** final state is painted blue, then the machine accepts some words. If not, then it does not accept any words.

*Option 1*

The algorithm has been followed correctly, and the correct conclusion has been reached. This is thus the correct option.

*Option 2*

The algorithm has not been followed correctly as only paths leading to the first final state were deleted. This is thus not the correct option.

*Option 3*

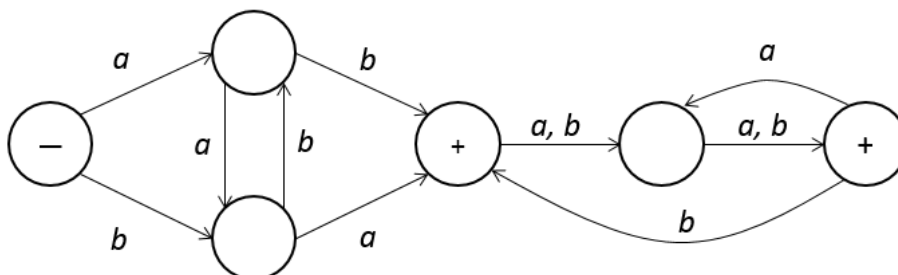
The algorithm has been followed partially, although not all edges from blue states were deleted. This is thus the correct option.

*Option 4*

The algorithm has not been followed correctly as not all paths were followed, only those leading directly to the first final state. This is thus not the correct option.

**Question 10**

Consider the following FA.



Which one of the following would be the correct decision procedure to determine whether this FA accepts a finite or an infinite language?

1. Consider the word  $w = aaaaaaa$  which is accepted by the FA.  
Now  $6 \leq \text{length}(aaaaaaa) < 12$   
Thus the language accepted is infinite.
2. Consider the word  $w = aaaaaa$  which is accepted by the FA.  
Now  $6 \leq \text{length}(aaaaaa) < 12$   
Thus the language accepted is infinite.
3. Consider the word  $w = bbbbbb$  which is accepted by the FA.  
Now  $6 \leq \text{length}(bbbbbb) < 12$   
Thus the language accepted is finite.
4. Consider the word  $w = ab$  which is accepted by the FA.  
Now  $0 \leq \text{length}(ab) < 6$   
Thus the language accepted is finite.

Answer: Option 1

#### *Discussion*

Theorem 19 on page 215 of Cohen presents an effective procedure for determining whether an FA accepts an infinite language or not.

Let  $F$  be an FA with  $N$  states.

Then, if  $F$  accepts an input string  $w$  such that  $N \leq \text{length}(w) < 2N$ , then  $F$  accepts an infinite language.

As in this case,  $N=6$ , we should have  $6 \leq \text{length}(w) < 12$ .

We use this to select the correct option.

#### *Option 1*

As the FA has  $N = 6$  states, and  $2N = 12$ , and  $aaaaaaa$  is a word with length 7 (and  $6 \leq 7 < 12$ ) which is accepted by the FA, this is the correct option to choose.

#### *Option 2*

This option is not correct as  $aaaaaa$  is not accepted by the FA.

#### *Option 3*

This option is not correct, Even though  $bbbbbb$  is in the language accepted by the FA, and  $6 \leq \text{length}(bbbbbb) < 12$  is true, the conclusion is wrong.

#### *Option 4*

This option is not correct as the theorem has been implemented incorrectly;  $N$  and  $2N$  were not used in the length comparison.