UNIVERSITY EXAMINATIONS

UNIVERSITEITSEKSAMENS

UNISA university of south africa

COS2601

October/November 2017

THEORETICAL COMPUTER SCIENCE II

Duration 2 Hours

100 Marks

EXAMINERS
FIRST MR CL PILKINGTON
SECOND DR WJC VAN STADEN

Closed book examination

This examination question paper remains the property of the University of South Africa and may not be removed from the examination venue

This examination question paper consists of 20 pages

Instructions

1 Answer all questions in the spaces provided on this exam paper

2 Extra pages are provided at the end of the exam paper for all rough work

3 The mark for each question is given in brackets next to the question

4 Unless otherwise specified, all languages in the questions are defined over the alphabet $\Sigma = \{a\ b\}$

ALL THE BEST!!

[TURN OVER]

QUESTION 1: Languages [10]

(a) Let S = {*ab bb*} and let T = {*ab bb bbb*}
(i) Is S* a subset of T*? Motivate your answer
(ii) Is S* = T*? Motivate your answer (4)

(i) Yes S* is a subset of T* since all the words in S* are found in T* but some words in T* are not in S* e.g bbb exists in T* not S*

(ii) S* ≠ T* since bbb is in T* but not S*

(b) Given S = {*ab abc cc*}, give all the words in S* that have length less than or equal to 4 (3)

ab, abc, cc, abcc, ccab, abab, cccc

(c) Given S = {*ab bab baab*}. Are the following words in S*? Indicate why or why not
(i) Λ
(ii) *abbabaabab* (3)

(i) Λ is in S* by definition of the kleene closure

(ii) (ab)(bab) aabab : the word abbabaabab is not in S* since the aabab substring cannot be formulated

QUESTION 2: Regular expressions **[10]**

(a) Give a regular expression that generates the language that consists of all words that start with a double letter (that is, two *a*'s or two *b*'s), end in a *b*, and have at least 4 characters (3)

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

(b) Can *abbaabaaabba* be generated by the regular expression **(a + b)(abb + baa)*aa***? Explain fully why you say so by showing how the word is, or is not, generated by the regular expression (2)

abbaabaaabba cannot be generated by the R.E since there is no way of generating bb-substring after the first letter

(c) Does the regular expression
a(aa + bb + ab + ba + aaa + abb + aab + aba + aab + bbb + baa + bab)*
define the language consisting of all words starting with an *a* and in which words of length 2 do not occur? If yes, explain why you say so, if no, give a counterexample (3)

No it does not since abba is in the language but cannot be generated by the regular expression

(d) Describe in words, in the simplest terms possible, the language generated by the regular expression **(a*b*)***. Provide another regular expression that generates the same language (2)

(a + b)*
Language of all words consisting of a, b including the null string

QUESTION 3: Recursive definitions **[10]**

A recursive definition for the language ODDA defined over the alphabet $\Sigma = \{a \ b\}$ should be compiled ODDA consists of all words of odd length that contain the *a*-substring

Provide

(a) an appropriate universal set, (1)
(b) the generator(s) of ODDA, (1)
(c) an appropriate function on the universal set, and then (1)
(d) use these concepts to write down a recursive definition for the language ODDA (7)

NOTE Marks will be subtracted for words that are introduced into the language that are not in ODDA

(a)

$\Sigma^{\infty} = \{a \ b\}^{\infty}$

(b)

a

(c)

CONCAT

(d)

ODDA $\subset \Sigma^{\infty}$

a $\in$ ODDA

If w $\in$ ODDA, then CONCAT(w, aa), CONCAT(w, ab), CONCAT(w, bb), CONCAT(w, ba), CONCAT(aa, w), CONCAT(ab, w), CONCAT(bb, w), CONCAT(ba, w) $\in$ ODDA

QUESTION 4: Mathematical induction [10]

(a) Provide a recursive definition for the set P of all integers greater than or equal to 0, (1)
(b) formulate the associated induction principle, and then (1)
(c) apply this induction principle to prove that $n^3 + 2n$ is divisible by 3 for all $n \geq 0$ (8)

Remember that to prove x is divisible by 3 you need to show that $x = 3y$ for some $y \in Z$

(a) P is a set of integer greater than 0 such that
1) $0 \in P$
2) If $x \in P$, then $(x+1) \in P$
3) No other element exist in P except for that described by 1st 2) above

(b) Let A be a subset of P such that
0 is a element of A. If $k+1 \in A$ each time an arbitrary element $k \in A$ then $k = A$

(c) Define $A \subseteq P$,
let A be a subset of P such that
$0 \in A$
Show for n= 0

$LHS = 0^3 + 2(0)$
$= 0$ (multiple of 3)

Assume $k \in A$ ie $k^3 + 2k = 3y$ where $y \in Z$

Required to prove $(k+1) \in A$ ie $(k+1)^3 + 2(k+1)$ is divisible by 3

LHS = $(k+1)^3 + 2(k+1) = k^3 + 2k^2 + k^2 + 2k + 3k + 3$
$= k(k^2 + 2k) + (k^2 + 2k) + 3k + 3$
but $k^2 + 2k = 3y$
$= k(3y) + 3y + 3k + 3$

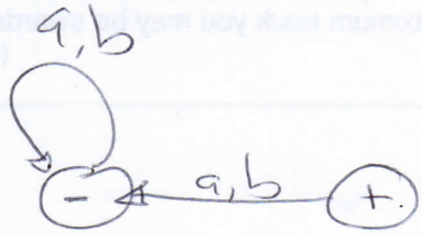
3ky, 3y, 3k and 3 are all divisible by 3 therefore their sum is also divisible by 3 meaning that LHS is divisible by 3

Thus

A = P since $k+1 \in A$ for all $k \in A$ and the statement $n^3 + 2n$ is divisible by 3 for all $n \geq 0$ holds

QUESTION 5: Finite automata **[10]**

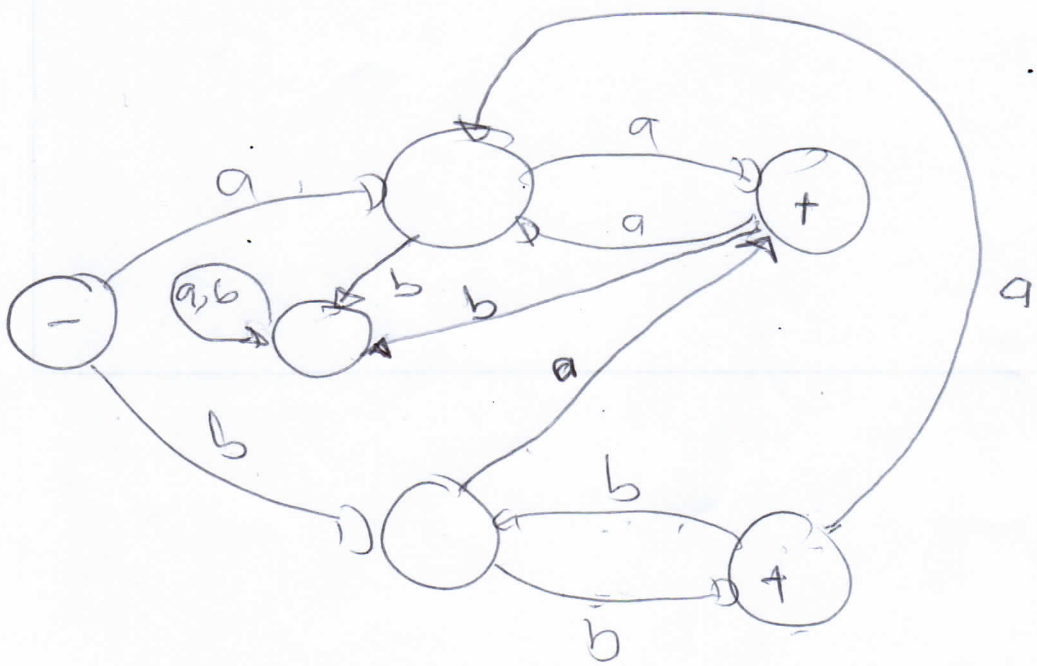
(a) Provide an example of a finite automaton (FA) that does not accept any words (2)



(b) Consider the language L, defined over the alphabet $\Sigma = \{a\ b\}$, consisting of all words that

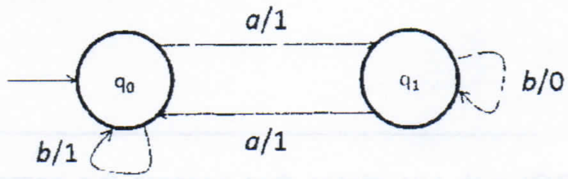
- are of even length, and
- do not contain the *ab*-substring

Build an FA (finite automaton) that accepts L Note that if you provide a nondeterministic finite automata (NFA) or transition graph (TG), the maximum mark you may be awarded is 2 Ensure that you build a deterministic FA (8)



QUESTION 6: Regular language acceptors **[10]**

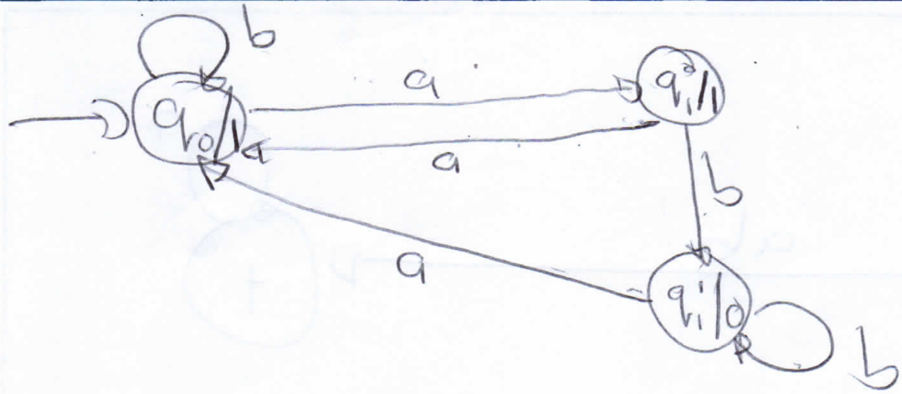
Consider the following Mealy machine



(a) What is the output if the input string *aabab* is run on the above Mealy machine? (1)

11110

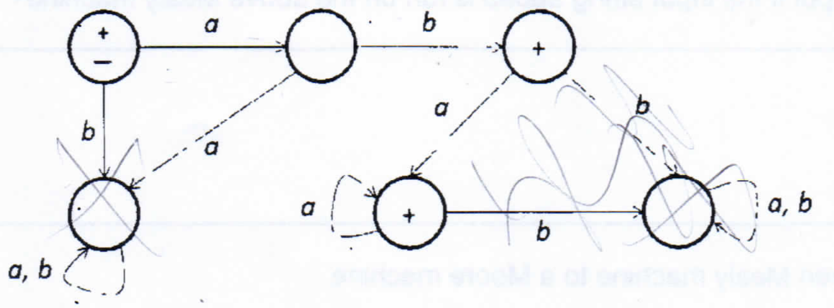
(b) Convert the given Mealy machine to a Moore machine (5)

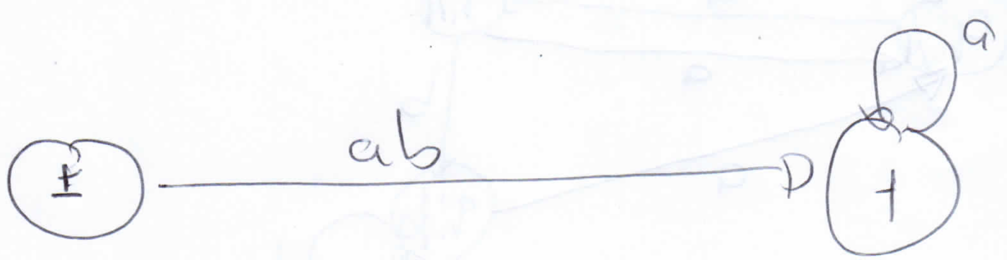


(c) What is the output if the input string *aabab* is run on the Moore machine from (b)? (1)

111110

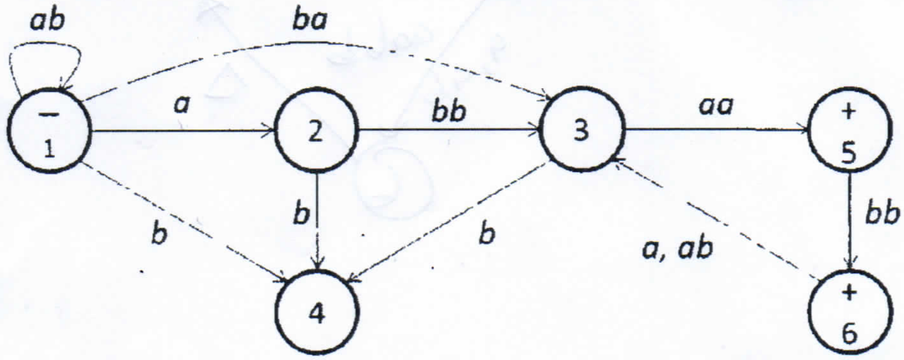
(d) Give a transition graph (TG) with two states that accepts the same language as the language accepted by the following finite automaton (FA) (3)



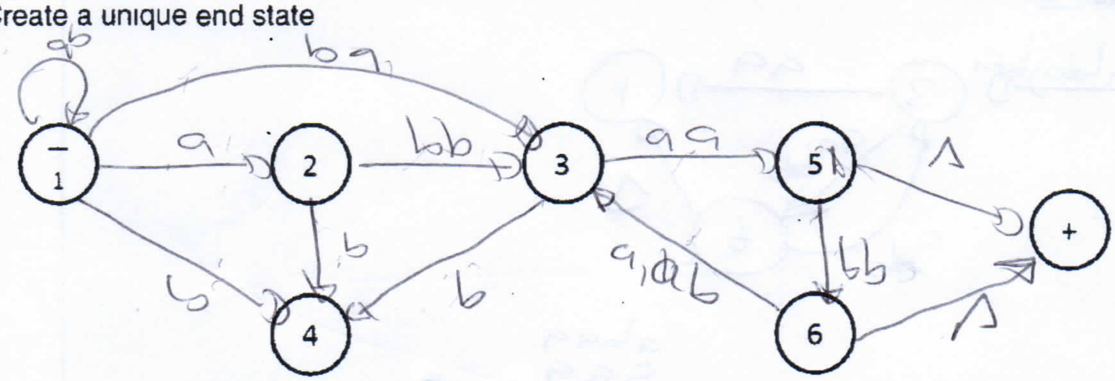


QUESTION 7: Kleene's theorem (TG to RE) [10]

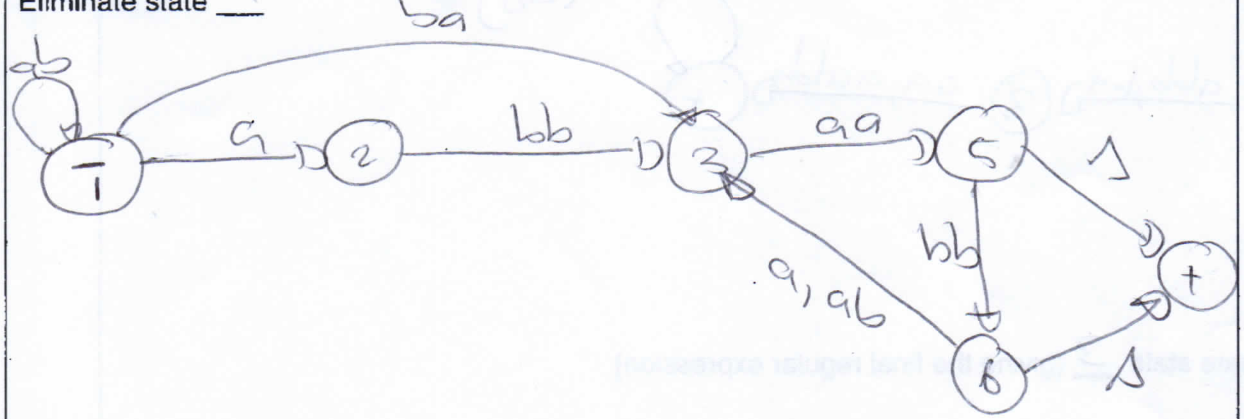
By applying Kleene's theorem, find a TG with 2 states that generates the language accepted by the following TG (10)



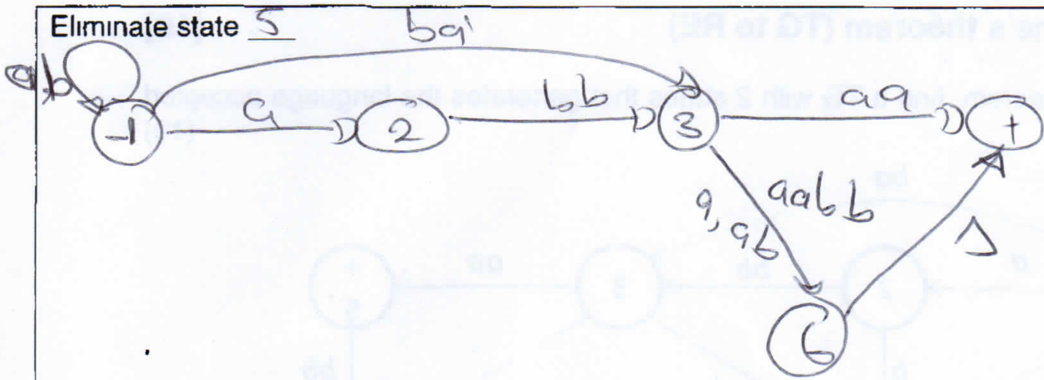
Create a unique end state



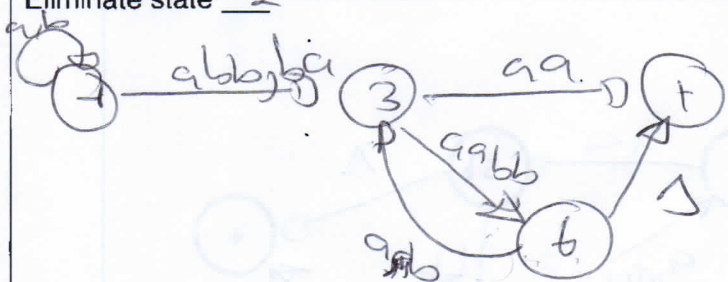
Eliminate state 4



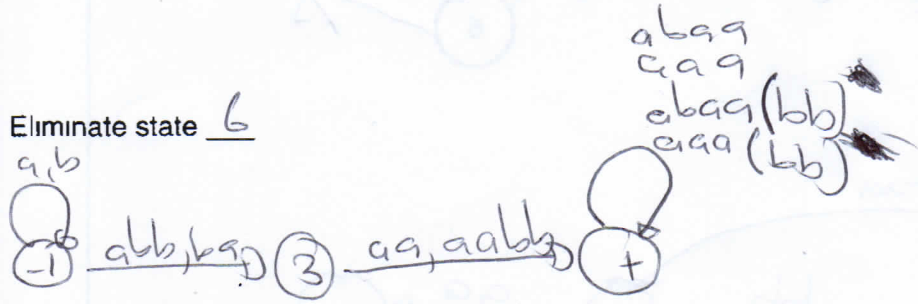
Eliminate state 5



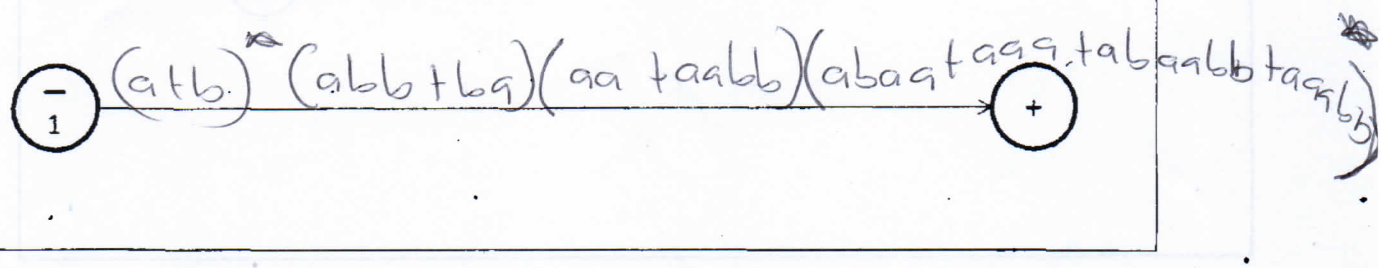
Eliminate state 2



Eliminate state 6

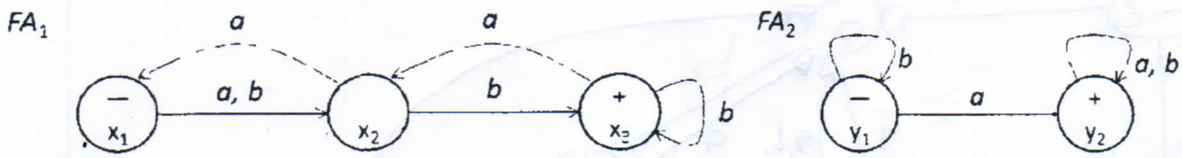


Eliminate state 3 (giving the final regular expression)



QUESTION 8: Kleene's theorem (RE to FA) [10]

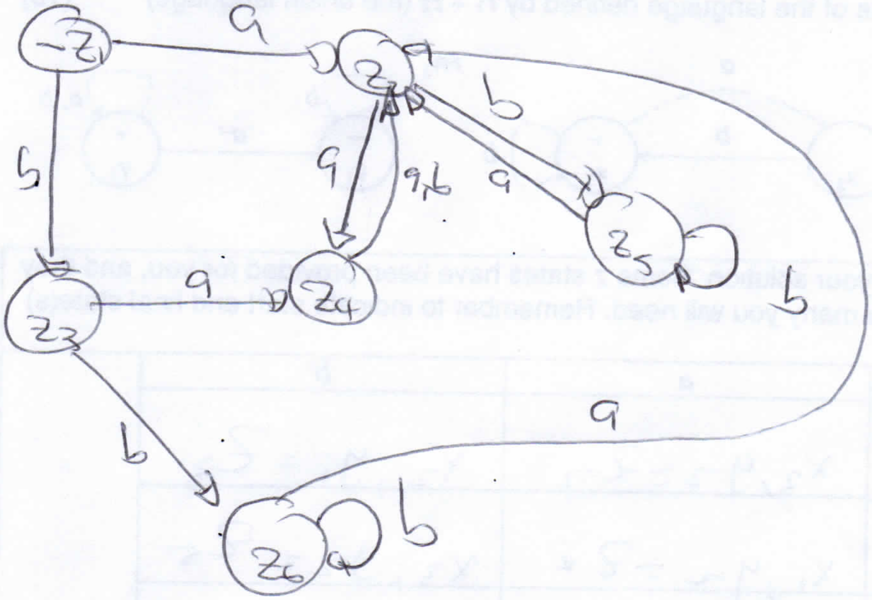
Let r_1 and r_2 be regular expressions and FA_1 and FA_2 be finite automata that accept exactly the languages defined by r_1 and r_2 respectively. By applying Kleene's theorem, build another FA that accepts all the words of the language defined by $r_1 + r_2$ (the union language) (10)



Use the table below to find your solution. Some z states have been provided for you, and they are not an indication of how many you will need. Remember to indicate start and final state(s)

New state	a	b
− z_1 = x_1, y_1	$x_2, y_2 = z_2$	$x_2, y_1 = z_3$
+ z_2 = x_2, y_2	$x_1, y_2 = z_4$	$x_3, y_2 = z_5$
z_3 = x_2, y_1	$x_1, y_2 = z_4$	$x_3, y_1 = z_6$
+ ~~z4 =~~ x_1, y_2	$x_2, y_2 = z_2$	$x_2, y_2 = z_2$
+ $z_5 = x_3, y_2$	$x_2, y_2 = z_2$	$x_3, y_2 = z_5$
+ $z_6 = x_3, y_1$	$x_2, y_2 = z_2$	$x_3, y_1 = z_6$

Now draw the new FA for $r_1 + r_2$



QUESTION 9: Pumping lemma with length [10]

Use the Pumping Lemma **with length** to prove that the following language is non-regular
L = { $a^{n-1}b^{n+1}c^n$, n ≥ 0 } (10)

Use the prompts below to complete the proof

Assume

L is Regular

Then there exists

an finite Automata with N-states and words of length greater than N

We choose any word w = $a^{k-2} b^{k+1} c^{k}$ where

k = N (Number of states)

Thus we can write w = xyz

for all length(w) > N

Then according to the pumping lemma with length, length(y) > 0

length(xy) ≤ N and such that $xy^nz \in L$

There is/are 5 possible choice(s) for y

1) y consists of a's only
2) y consists of b's only
3) y consists of c's only
4) y consists of ab-substring
5) y consists of bc-substring

If y is pumped in each of the above case(s),

for case 1-3 after pumping, there is an uneven ratio of a's b's and c's

for case 4-5 after pumping there will be more than 1 instance of the ab or bc substring respectively

We thus conclude that

there is no possible choices for y where $xy^nz \in L$

And

Thus we retract our assumption and conclude that L is not regular according to the pumping lemma with length

QUESTION 10: Decidability **[10]**

(a) Use the algorithm that eliminates symbols from a regular expression in steps to show that **b(aa + bb)***(**ab**)* + (Λ + **a***) accepts at least one word (3)

b(aa+bb)* (ab)* + (Λ+a*)
b(aa+bb)(ab) + (Λ+a)
b(aa)(ab) + Λ
baaab

(b) State whether the following are true or false
(i) There is an effective procedure to decide whether a given FA accepts any words
(ii) There is not an effective procedure to decide whether two FAs are equivalent (2)

(i) TRUE

(ii) FALSE

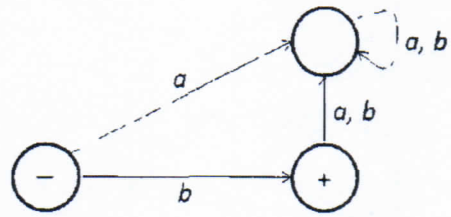
(c) Let F be an FA with N states State the theorem or test that can be used to determine whether F accepts an infinite language (2)

If F accepts a word w, such that

$N \leq length(w) < 2N$

then F accept an infinite language

(d) Using the theorem or test given in (c), determine whether the following FA accepts an infinite language (3)



FA only accepts one word which is b. Therefore since ~~we~~ there are no word of length between 3 and 6, ~~the~~ ~~langua~~ the FA therefore accept a finite language.

ROUGH WORK
Work done here will not be marked