National University of Computer and Emerging Sciences, Lahore Campus

Course:	Theory of Automata	Course Code:	CS 301
Program:	BS(Computer Science)	Semester:	Fall 2016
Duration:	180 Minutes	Total Marks:	55
Paper Date:	28-DEC-16	Page(s):	9
Section:	A,B,C,D	Section:	
Exam:	Final	Roll No:	

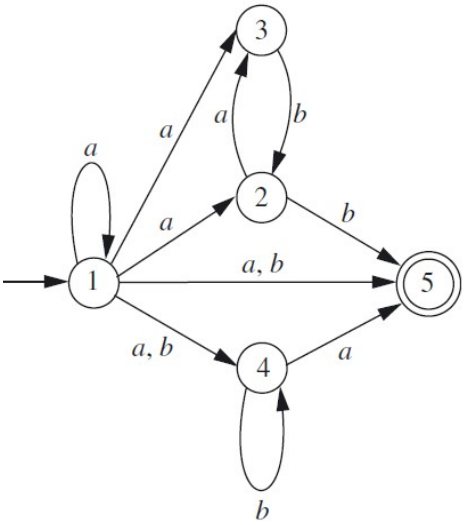
Instruction/Notes:

1. All students can bring **one hand written A4 help sheet**in the exam hall
2. Sharing of A4 sheet is strictly **NOT ALLOWED**
3. Answer in the space provided
4. You can ask for rough sheets but they will not be graded or marked

Good luck!

QUESTION 1 (Marks:5)

Convert the given NFA to DFA



Solution:

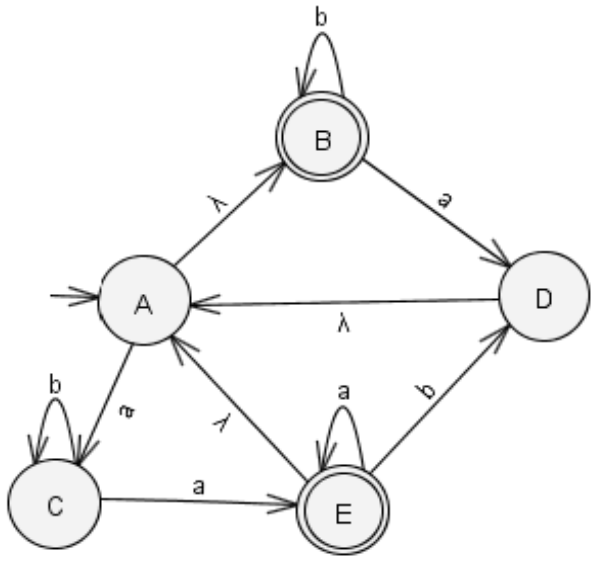
state	a	b
1	123 45	45
123 45	123 45	24 5
45	5	4
245	35	45
5	Trap	Tra p
4	5	4
35	trap	2
2	3	5
3	trap	2
Trap	Trap	tra

Roll Number:_____ Section: _____

		p

QUESTION 2 (Marks:1 +4)

Given the NFA-null



a. Using the extended transition function of an NFA-null find out if the string ***abab*** is part of the language or not.

b. Convert the above NFA-null to NFA

Solution:

a. {A,B,D}

b.

State	a	b
A	{A,B,C, D}	{B}
B	{A,B,D}	{B}
C	{A,B,E}	{C}
D	{A,B,C, D}	{B}
E	{A,B,C, D,E}	{A,B, D}

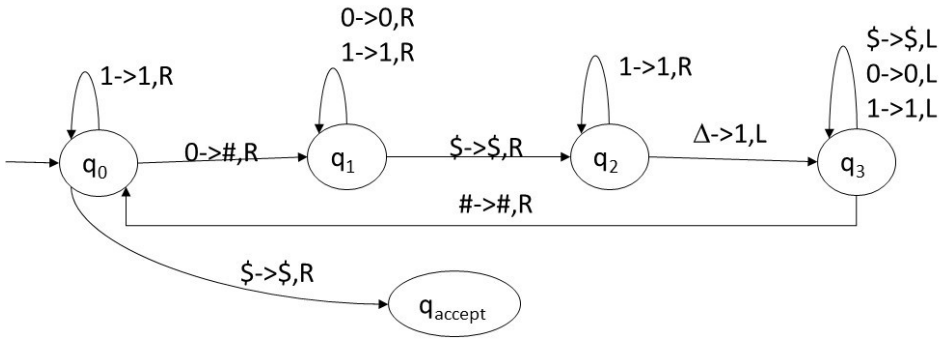
Roll Number:_____ Section: _____

Roll Number:_____ Section: _____

QUESTION 3 (Marks: 1+1+1+1+2)

You are given a single tape TM.Blanks are represented by Δ. The edges are represented by the notation:

Current Symbol->Replaced symbol, Direction



a. You have to run the TM on the following strings and write the output on the tape when the machine halts.

Input on tape	**Output on tape**
101$	**1#1$1**
100010$	**1###1#$1111**
111$	**111$**
0000$	**####$1111**

b. What does the above machine do? (answer should be generic).

It count number of 0's in the given input

QUESTION 4 (Marks: 1+4)

Suppose we are given three strings ϵ {a,b,c}* as input, we have to determine the index of the string which has maximum length.

Example: if String1=abd String2=cccc and String3=cbabbb are the input, the output= 3

a. Write an algorithm, in plain English, to solve this problem using a TM.

b. Create a single tape deterministic Turing machine for your algorithm. Initially assume that all three input strings are separated by the delimiter #. Also, the input starts with # symbol. So when given the input on tape as:

#abd#cccc#cbabbb#, the tape when TM halts should be: #abd#cccc#cbabbb#3

Roll Number:_____ Section: _____

Idea: Compare String 1 and 2, whichever string is larger in this comparison compare it with 3, which ever string is larger in this comparison write its index on end of tape. If in any comparison both strings are of equal length, one will be selected randomly.

Algorithm:

Step 1: Look for 1st uncrossed symbol of String 1,
If found cross it go to step 2
If not found, go to step 3 // string 2 is larger or equal than string 1

Step 2: Look for 1st uncrossed symbol in String 2,
If found cross it and go to step1
If not found, go to step 4 // string 1 is larger or equal than string 2

Step 3: // string 2 is larger or equal than string 1.
Uncross all the symbols of String 2//restore string 2 compare with String 3

Step 3b: Look for 1st uncrossed symbol in String 2,
If found cross it and go to step3c
If not found; go to end of tape and write 3, halt. // string 3 is greater of equal to string 2 and 1

Step 3c: Look for 1st uncrossed symbol in String 3,
If found cross it and go to step3b
If not found go to end of tape and write 2, halt. // string 2 is greater of equal to string 3 and 1

Step 4: // string 1 is larger or equal than string 2.
Uncross all the symbols of String 1 //restore string 1 compare with String 3

Step 4b: Look for 1st uncrossed symbol in String 1,
If found cross it and go to step4c
If not found; go to end of tape and write 3, halt. // string 3 is greater of equal to string 1 and 2

Step 4c: Look for 1st uncrossed symbol in String 3,
If found cross it and go to step4b
If not found go to end of tape and write 1, halt. // string 1 is greater of equal to string 3 and 2

Roll Number:_____ Section: _____

QUESTION 5 (Marks: 3+3+3+3)

a. Using induction prove that any string s $\in \{0,1\}^*$ can be written as s = xy such that $n_0(x) = n_1(y)$, where $n_a(z)$ denotes the number of occurrences of the symbol a in the string z. |x| >= 0 and |y| >= 0

Step 1: Base Case: for s= null
X=null y =null
S= XY=null
$N_0(X)=N_1(Y)$

Step 2: Let the given statement be true for s'
|s'|=k
X=0 , Y=1

$S'=X^kY^k = 0^k1^k$
Such the n0(X)=n1(Y)

Step 3: For s'' of length K+1
If
$S''= X^{k+1}Y^{k+1} = 0^{k+1}1^{k+1}$
$S''= XX^kY^kY = 00^k1^k1$
S''= X S'Y = 0 S'1
S'' = S' as |X|=|0|=|Y|=|1|=1

Proved n0(X)=n1(Y)

Roll Number:_____ Section: _____

Is the following statement correct for any two languages L_1 and L_2? Give a proof:

$(L_1L_2)^* = L_1^*L_2^*$

Dis-proof by contradiction:

Let L1= {0}
Let L2= {1}

{L1L2}*= {01}*= {null, 01,0101,010101,....}
L1*={null,0,00,000,....}
L2*= {null, 1, 11, 111,....}
L1*.L2*= {null, 01,001,0011,....}

0101 is in $(L_1L_2)^*$ but not in $L_1^*L_2^*$ so $(L_1L_2)^* != L_1^*L_2^*$

b. Give a regular expression for all binary strings so that there is an even number of 0's between any two 1's, e.g., the strings 10010000111001 and 110010000001

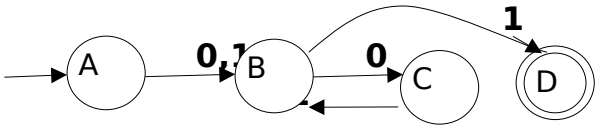
belong to the language. (you can consider zero as an even number). The strings 101 and 10001 do not belong to the language.

RE: (1* + 0* + 1(00)*1)*

c. Convert the given regular expression into an NFA-null

(0+1)(01)*1

Solution:



Roll Number:_____ Section: _____

QUESTION 6 (Marks: 4+4)

a. Convert the following CFG to a PDA:

S ->aB | bA
B -> Cab | ε
C ->aa | bb | aaC | abC | ε
A -> aba | aab | ε

Solution:

State	Input	Top of Stack	Move
Q1	-	Z_0	Q2,Z_0S
Q2	-	S	(Q2,aB), (Q2,bA)
Q2	-	A	(Q2,aba), (Q2,aab), (Q2,ε)
Q2	-	B	(Q2,Cab), (Q2, ε)
Q2	-	C	(Q2,aa),(Q2,bb) , (Q2,aaC) ,(Q2, abC) ,(Q2,ε)
Q2	A	a	Q2,pop
Q2	-	Z_0	Q3, Z_0

b. Make a deterministic PDA for the following language L.

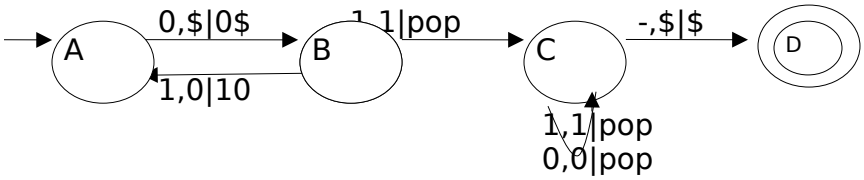
L = $\{(01)^n(10)^n \mid n > 0\}$

Assume that the initial stack has a $ placed on it. You have to represent each edge with one of the following notations BUT NOT BOTH:

input,stack_top_character / string_that_replaces_stack_top

OR

input, pop character x --> push character y



Roll Number:_____ Section: _____

QUESTION 7 (Marks: 3+5+3+3)

a. Show that if any language L is decidable then the complement of that language L' is also decidable

If the language L is decidable then there must be some Turing machine M that will decide it, that is for every input x it will either accept or reject.
Using M we can create a decider N of L' as well that will work as follow.

N= "For input x
1. Simulate M for x
2. If M accepts, reject
3. If M rejects accept

b. Given that L_1 is undecidable and Turing recognizable:
L_1: (B | B is an ambiguous CFL)
Describe an algorithm for recognizing L_1
Now given that if a language is undecidable, its compliment is Turing Unrecognizable. **Is L_2 decidable, undecidable or unrecognizable?**
L_2: (B | B is an unambiguous CFL)

If L1 is Turing recognizable then there is a Turing Machine M that can recognize it. Following is the description of M

M: "Input <B> where B is CFL
For every x ϵ ∑* (terminals)
Find every possible parse tree to derive x from <B>
If there are more than 2 parse trees of x; accept "
As we can see that there are infinite number of possible x , therefore reject state will nevere be reached. This makes M a recognizer of L1

L2 is unrecognizable:
Proof by contradiction:
Let L2 be recognizable. As L2 is complement of L1, if both L1 and it complement are recognizable then L1 is decidable. This is a contradiction from the given statement that L1 is undecidable. Therefore our assumption that L2 is recognizable is false, making L2 unrecognizable.

c. Euclidean Algorithm to decide if the Greatest Common Divisor (GCD) of two numbers a and b is c, i.e, decide if GCD(a,b) = c

```
Step 1: if a=b goto step 3
Step 2: if a>b
            a:=a-b
    else
            b:=b-a
    goto step1
Step 3:  GCD= a
    if (a==c) accept else reject
```

Does the above algorithm belong to P class? Give reason

Yes, the given algorithm is in P class as the maximum number of iterations will be N= max (a,b). The algorithm can be simulated in Deterministic Turing machine in polynomial time.

Does the above algorithm belong to NP class? Give reason.

d. We have a set X of n numbers. Determine if X has a subset, such that the GCD of all numbers in that subset is 6. So for example if X= {1 3 4 6 8 10 20 18 36 42 48 60}, {6, 8, 60} is subset that has GCD 6. And X= {1, 2, 3, 4, 5, 7, 8, 13} has no such subset.

Does the above algorithm belong to P class? Give reason. Does the above algorithm belong to NP class? Give reason.