# Test 1

**Total: ___________/ 150**

**Printed Name:** ____Demetrius Johnson_______3-8-2020______

**GRADER/TA:** ***[CO for Question 1, P v NP, please record statistics]***

I, ________________________, have neither given nor received assistance on this examination except that which is provided by, or approved by, the instructor.

1. [15 pts] If an algorithm A is in the complexity class P, is it necessarily in the complexity class NP? Explain why or why not.

Well, here is the million dollar question: is $P \leq NP$? But, for this question, we do in fact know that P is a subset of P, it is only a matter of is P a proper subset of NP. It could be that NP problems have problems that do not belong to class P.

Short Answer: $\boxed{\text{yes}}$ If A is of class P, it is of class NP.

If the case were A is class NP, we could not necessarily say it is a class P problem. So $P \subset NP$, but NP is not necessarily in set P: $NP \underset{?}{\subset} P$
If it were so, then $P \leq NP \Rightarrow NP \leq P$.

2. [15 pts] Given an alphabet Σ = {a, b, c}, list all strings from length 1 to 3 in the language

**L = a(a + b + c)*** Assume the empty string λ is length 0.

~~$L = \lambda$~~

~~$L = a \cdot (a+b+c)$~~

~~$L = \lambda + a(a+b+c)^1 +$~~

~~$L = a(a+b+c)^0 (a+b+c)^1 (a+b+c)^2 \ldots$~~

~~$(a+b+c)^0 = \lambda$~~

$L = a[(a+b+c)^0 + (a+b+c)^1 + (a+b+c)^2 \ldots]$

$(a+b+c)^0 = \lambda$

$(a+b+c)^1 = a+b+c$

$(a+b+c)^2 = (a+b+c)(a+b+c) = a^2 + ab + ac + ba + b^2 + bc + ca + cb + c^2$

Solution: $\boxed{a(a+b+c)^0 = \cancel{\lambda}}$ — Length = 0

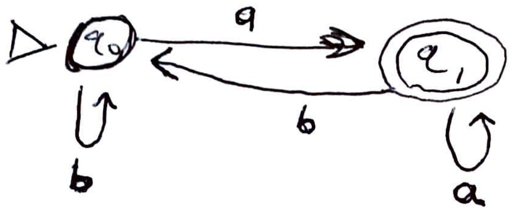
$a(a+b+c)^1 = a^2 + ab + ac$

$a(a+b+c)^2 = a^3 + a^2b + a^2c + aba + ab^2 + abc + aca + acb + ac^2$

Strings Length 1–3

3. [15 pts] Draw a state transition diagram for the FA of regular language L(r) where

**r = (a + b)*a** ⟶ Any Combinations of a's and b's including λ, ending with a



Demetrius Johnson

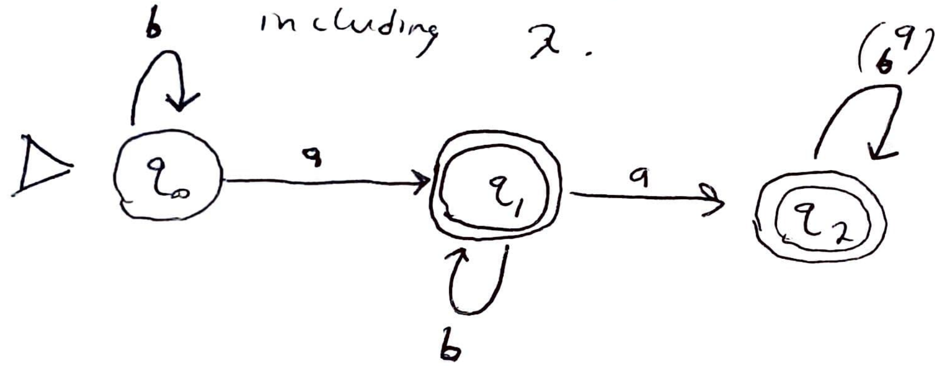
4. [15 pts] Write a regular expression for the language L using alphabet {a, b} where all strings in the language start with a single b, followed by any number of a's (including none), and end in a single a.

$L(r) = b a^* a$

5. [20 pts] Draw the state transition diagram for the finite automaton that models the regular expression **b*ab(a + b)***

~~Any # of bs (start)~~

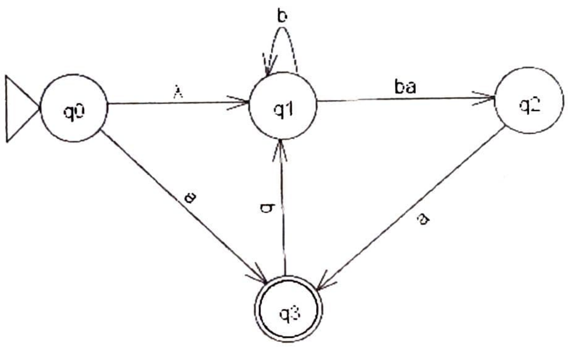
(1) start w/Any # of b's, including λ.
(2) followed by ab; string must contain ab.
(3) followed by any combination of a's and b's including λ.



Demetrius Johnson

6. [20 pts] Convert the following transition graph into a finite automaton by filling in the (final) table and **identifying** the start and accept states. You do not have to draw the state transition diagram for the FA, just fill in the table (note that this is NOT the intermediate table.

| FA State | a | b |
|---|---|---|
| 0 | 3 | (1,1.21) |
| (1,1.21) | 2 | 1 |
| 2 | 3 | BH |
| 3 | BH | (1,1.21) |
| | | |
| | | |
| | | |
| | | |




| state | a | b |
|---|---|---|
| 0 | 3 | 1, 1.21 |
| 3 | crash/BH | 1 |
| 1 | ~~crash/BH~~ crash/BH | 1 |
| (1, 1.21) | 2 | 1 |
| 2 | 3 | BH |

Combine these states

Demetrius Johnson

7. [15 pts] Give the configuration after applying the appropriate transition function, using the symbols a, b, c. Only apply the transition function *once*.

Assume the **original configuration is:** $\boxed{\textbf{bbabbq}_2\textbf{aaab}}$

**Available transition functions:**

- **$\delta(q_1, a) = (q_2, a, R)$**
- **$\delta(q_1, b) = (q_3, c, L)$**
- **$\delta(q_2, a) = (q_2, b, L)$**
- **$\delta(q_2, b) = (q_3, c, R)$**
- **$\delta(q_3, a) = (q_4, c, R)$**
- **$\delta(q_3, b) = (q_2, a, L)$**

R, W, Move

State: 2

Reading: a

Write: a, b, or c

Move: $\boxed{\text{Left}}$ or right

move Left

Write a "b"

Appropriate transition Function:

$\delta(q_2, a) = (q_2, b, L)$ → to bbabb$q_2$aaab

stay in state 2.

Yields: $\boxed{bbab\underline{q_2}bbaab}$

8. [15 pts] Given sets A = {a, b, c, f, g} and B = {a, c, d, g}:

a. Find A ∪ B

$\boxed{A \cup B = \{a, b, c, d, f, g\}}$

b. Find A ∩ B

$\boxed{A \cap B = \{a, c, g\}}$

c. Find A - B

$A - B = \{a, b, c, f, g\} - \{a, c, d, g\}$

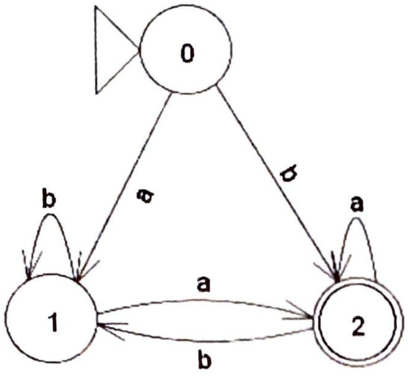
$\boxed{A - B = \{b, f\}}$

Final Solution for #9:

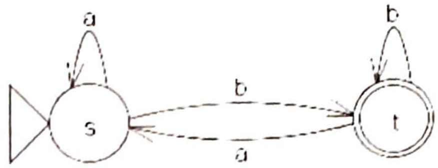
$L_1' \cup L_2$ Accept states = (1, t)


Prof. John P. Baugh, Ph.D.
Demetrius Johnson

9. [20 pts] Name the accept states (accept *group* states) in the machine $L_1' \cup L_2$




$L_1$




$L_2$

Accept states = {2, t}
(Both)

Reject states = {1, 0, s}
(Both)

Start states = {0, s}
(Both)

| group State | a | b |
|---|---|---|
| (0, s) | (1, s) | (2, t) |
| (1, s) | (2, s) | (1, t) |
| (2, t) | (2, s) | (1, t) |
| (2, s) | (2, s) | (1, t) |
| (1, t) | (2, s) | (1, t) |

$L_1'$ = NonAccept states = {1, 0}

$L_2$ = Accept states = {t}

Accept states for machine

Solution:

Accept states for → $L_1' \cup L_2$ = (1, t)

No Accept

Accept