CSCI 338, Final Exam – May 5th, 2016

Name SAMPLE SOLUTION

Question One. 20 points. Let 4SAT = {<Φ> | Φ is a satisfiable 4cnf-formula}.

Part A. 5 points. Show an instance of 4SAT that has 2 clauses over the variables a, b, c and d. Recall that a clause consists of several literals connected with ORs.

$(a \vee b \vee c \vee d) \wedge (\bar{a} \vee \bar{b} \vee c \vee d)$

Part B. 5 points. Demonstrate that 4SAT is a member of the NP class.

A set of truth assignments can be verified that all clauses are true in polynomial time

Part C. 5 points. Show how to reduce the 3SAT problem below to an instance of the 4SAT problem in polynomial time.

3SAT Problem	4 SAT Problem
(a v b v c) & (¬c v d v e)	$(a \vee b \vee c \vee c) \wedge (\neg c \vee d \vee e \vee e)$

Part D. 5 points. What is the most important fact about 4SAT that Parts B and C tell us?

4SAT is an NPC problem

Question Two. 15 points. Convert the following Boolean expression into an instance of the 3SAT problem:

¬(a & b & c & d) v (e & f)

$$\equiv (\neg a \lor \neg b \lor \neg c \lor \neg d) \lor (e \land f)$$

$$\equiv (\neg a \lor \neg b \lor \neg c \lor \neg d \lor e) \land (\neg a \lor \neg b \lor \neg c \lor \neg d \lor f)$$

$$\equiv \left.\begin{aligned} &(\neg a \lor \neg b \lor x_1) \land \\ &(\neg x_1 \lor \neg c \lor x_2) \land \\ &(\neg x_2 \lor \neg d \lor e) \land \end{aligned}\right\} \text{from first clause}$$

$$\left.\begin{aligned} &(\neg a \lor \neg b \lor y_1) \land \\ &(\neg y_1 \lor \neg c \lor y_2) \land \\ &(\neg y_2 \lor \neg d \lor f) \end{aligned}\right\} \text{from second clause}$$

Question Three. 10 points. Consider a binary search tree that contains n items. For parts A-G, fill in the blank with the most appropriate asymptotic notation: w, Ω, Θ, big-O, or little-o. If no notation is appropriate, write XXX. For parts H-J, circle the correct answer.

Part A. Searching for an item that is present is Ω(1)

Part B. Searching for an item that is present is XXX(log n)

Part C. Searching for an item that is present is big O(n)

Part D. Searching for an item that is present is little o(n log n)

Part E. The best case of searching for an item that is NOT present is little o(log n)

Part F. The average case of searching for an item that is NOT present is XXX(log n)

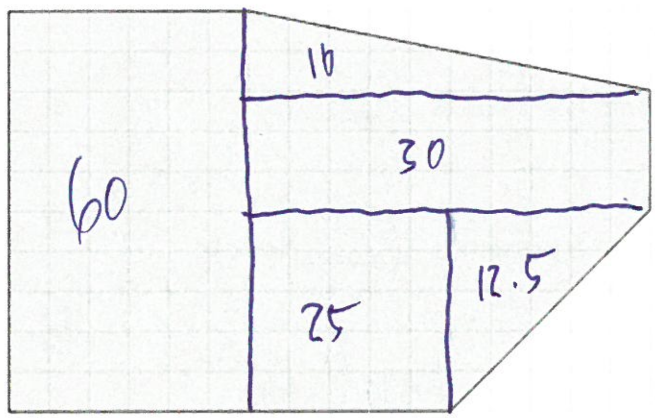
Part G. The worst case of searching for an item that is NOT present is w(log n)

Part H. True or False. An algorithm that is O(n^2) must be O(n^3).

Part I. True or False. An algorithm that is Θ(n^2) must be Θ(n^3).

Part J. True or False. An algorithm that is w(n^2) must be w(n^3)

Question Four. 10 points. The problem of finding the area of the hexagon in the unit squares below can be reduced to five simpler problems. Show and solve those simpler problems.



area = 137.5

Question Five. 10 points. Recall that A_{TM} = {<M, w> | M is a Turing Machine and M accepts w} is undecidable. Complete the proof below to show that R_{TM} = {<M, w> | M is a Turing Machine that rejects w} is undecidable.

Proof by Contradiction: Assume R_{TM} is decidable. We can therefore construct Turing Machine X to decide it. Construct Turing Machine Y to decide A_{TM} as follows:

Y = "On input <M, w> where M is a Turing Machine and w is a string:

1. Run X on <M, W>
2. If X rejects, Y accepts
3. If X accepts, Y rejects"

If X decides R_{TM} then Y decides A_{TM}.

Since A_{TM} is undecidable, R_{TM} is undecidable

Question Six. 15 points. Consider the Turing Machine transition diagram below.

Part A. What is Σ?

{0, 1, #, $}

Part B. What is Γ?

{0, 1, #, $, ␢}

Part C. What is Q?

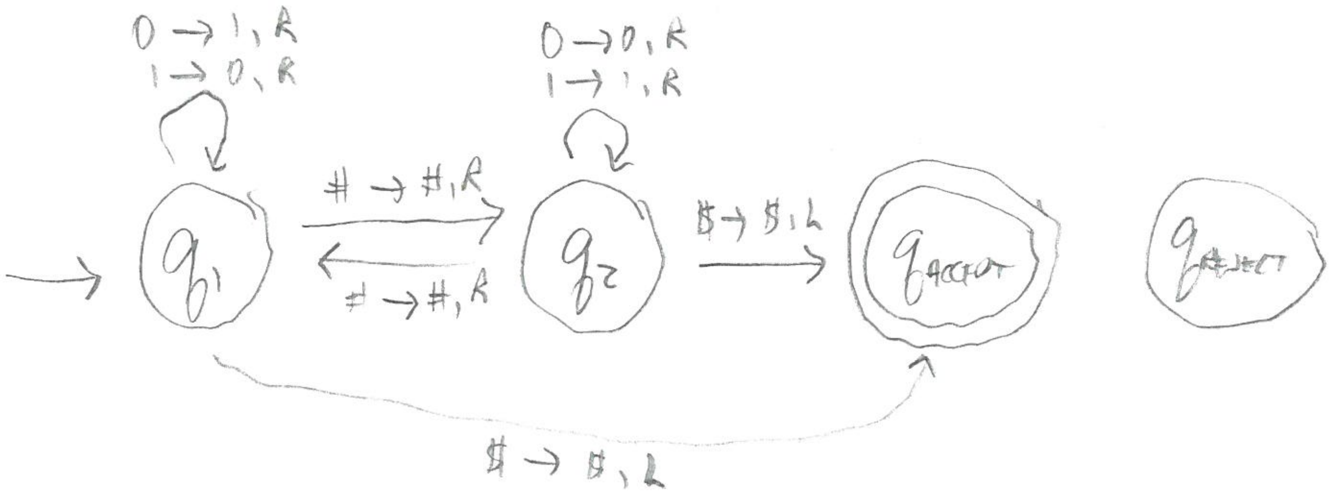
$\{q_1, q_2, q_{ACCEPT}, q_{REJECT}\}$

Part D. Show the contents of the tape at the end of the computation if the tape initially contains **0101#0101#0101#$**

1010 # 0101 # 1010 # $

Part E. Describe briefly the computation that is performed when the initial contents of the tape are **binary-number-1#binary-number-2# … #binary-number-k#$**

Binary numbers in odd positions are inverted (e.g. 0 becomes 1, 1 becomes 0)



Question Seven. 10 points. Consider the language A = $\{a^i b^j c^{i+j} \mid i, j >= 0\}$. Consider the following pumping lemma proof:

Assume that A is regular. Let p be the pumping length given by the pumping lemma. Consider string = $a^p b^p c^{2p}$. By the pumping lemma, we know we can split s into three pieces, xyz, where for any i >= 0 the string xy^iz is in A. We know that |y| >= 1. Consider the five cases when i = 0:

1. If y contains only a's, the number of a's and b's is now less than the number of c's.
2. If y contains only b's, the number of a's and b's is now less than the number of c's.
3. If y contains only c's, the number of a's and b's is now greater than the number of c's.
4. If y contains a's and b's, the number of a's and b's is now less than the number of c's.
5. If y contains b's and c's, the number of a's and b's is now greater than the number of c's.

A contradiction is reached. Therefore, A is not regular.

Part A. Identify the error in the proof above.

#5 is not true if y contains the same number of b's and c's.

Part B. Fix the error in the proof above.

#5 Consider xy^2z. $a^p b^p c^k b^j c^{2p} \notin A$

Question Eight. 10 points. Circle your answer.

a) True or False. A Finite State Automaton (FSA) has finite memory.
b) True or False. A Push Down Automaton (PDA) has finite memory.
c) True or False. A Linear Bounded Automaton (LBA) has a finite memory.
d) True or False. A Turing Machine (TM) has finite memory.
e) True or False. A nondeterministic FSA is more powerful than a deterministic one.
f) True or False. A nondeterministic PDA is more powerful than a deterministic one.
g) True or False. A nondeterministic TM is more powerful than a deterministic one.
h) True or False. A deterministic, single-tape TM captures the notion of an algorithm.
i) True or False. If Problem A, an NPC problem, can be reduced to Problem B, a P problem, in an exponential amount of time, then P = NP.
j) True or False. If Problem A can be reduced to an undecidable problem in a polynomial amount of time, then Problem A is an undecidable problem.