MIDTERM EXAM RULES AGREEMENT

Please read the rules of the exam and sign and date at the end of the document.

Here are the rules of the take-home exam. Please carefully read the instructions below. Failure to comply with any of the instructions below may result in our being unable to accept or grade your exam or initiating disciplinary actions.

1) You will have 24 hours to complete the second Midterm exam from 5:00 pm on Tuesday, April 14, until 5:00 pm on Wednesday, April 15.

2) This exam is "open book" which means you are permitted to use any materials handed out in class (like the scanned notes), your notes from the course, the textbook (Sipser's second edition), and anything on the course canvas.

3) The exam must be taken completely alone. Showing it or discussing it with anyone is forbidden.

4) You may not consult any external resources. This means no internet searches, materials from other classes or books or any notes you have taken in other classes, etc. You may not use Google or any other search engines for any reason. You may not use any shared Google documents.

5) You may not consult with any other person regarding the exam. You may not check your exam answers with any person. You may not discuss any of the materials or concepts in COT 5310 with any other person.

6) You must scan the answers of all the points in the exam and uploaded the exam to Canvas as a single pdf document. The pdf should be readable, try to have the best quality in the scan as possible.

I agree with the rules of the midterm exam.

Signature: --

Date: 04-15-2020

Florida International University
School of Computing and Information Sciences

Second Midterm Exam

COT 5310 Theory of Computation I
Spring 2020

Name: Jake Lopez
Netid: 5820326

- Upload the solutions to canvas a scanned with as much resolution as you can.
- Suggestions: Read through the entire exam first before starting work.
- If you run short on time, remember that partial credit will be given.
- If any question is unclear, ask for clarification. I will setup office hours during the exam period.

Question	Points	Score
Problem 1	10	
Problem 2	10	
Problem 3	10	
Problem 4	10	
Problem 5	15	
Problem 6	10	
Problem 7	10	
Problem 8	10	
Problem 9	10	
Problem 10	10	
Total	100	

1. True or False(10 points)

Completely write out "True" if the statement is necessarily true. Otherwise, completely write "False". Include a justification.

(a) If L is regular and L' is decidable, then $L \cap L'$ is decidable

True. Regular Languages are decidable and intersection is a closed operation.

(b) The complement of A_{TM} is Turing recognizable.

False. ATM is undecidable and recognizable. If ATM were recognizable ATM would have to be decidable which is a contradiction.

(c) If A reduces to B, and A is decidable, then B is decidable.

False. If B is decidable and A reduces to B then A would be decidable not the other way.

(d) A non-deterministic TM can decide languages that a regular TM cannot decide.

True. Non-deterministic TM's and regular TM's are equivalent.

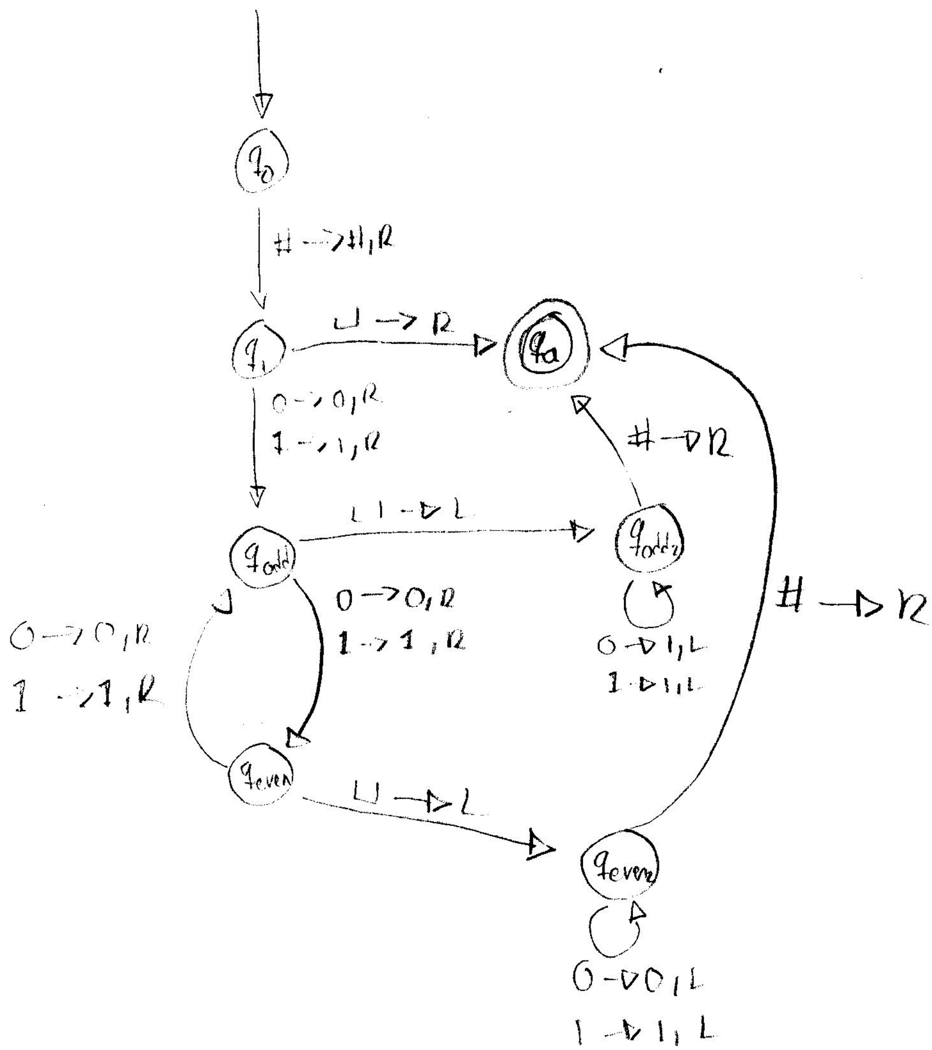
(e) If a language A is Turing-decidable, then its complement is Turing decidable.

True. By Sipser theorem 4.22 a language is decidable only when both the language and it's compliment are decidable.

2. Turing Machine Design (10 points)

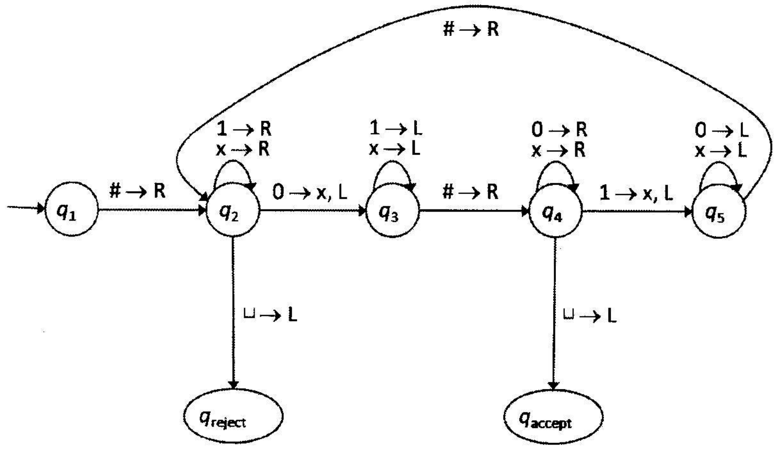
Give the state diagram of a TM M that does the following on input $\#w$ where $w \in \{0,1\}^*$. Let $n = |w|$. If n is even, then M converts $\#w$ to $\#0^n$. If n is odd, then M converts $\#w$ to $\#1^n$. Assume that ϵ is an even length string.

The TM should enter the accept state after the conversion. We do not care where you leave the head at the end of the conversion. The TM should enter the reject sate if the input string is not in the right format. However, your state diagram does not need to explicitly show the reject state or transitions into it.



3. TM notation and design (10 points)

A a Turing machine M is given as a 7-tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$ where $Q = \{q_1, q_2, q_3, q_4, q_5, q_{accept}, q_{reject}\}$, $\Sigma = \{0, 1, \#\}$, and $\Gamma = \{0, 1, \#, x, \sqcup\}$. The state diagram of the Turing M is shown as follows.



(a) (6 points) Write out the sequence of configurations that M enters when started on the input string #010

q_1 #010
#q_2 010
q_3 #x10
#q_4 x10
#x q_4 10

#xx q_5 0
#x q_5 x0
q_5 xx0
q_5 #xx0
q_2 xx0

#x q_2 x0
#xx q_2 0
#x q_3 xx
q_3 xxx
q_3 #xxx

q_4 xxx
#x q_4 xx
#xx q_4 x
#xxx q_4 ⊔
#xx q_{accept} x
halt.

(b) (4 points) Is $\langle M, \#1100\rangle \in A_{TM}$?

NO. M only accepts strings where the number of 0's is greater than the number of 1's.

 Name: Jake Lopez

4. Decidability (10 points)

Let $A = \{\langle R, S\rangle \mid R$ and S are regular expressions and $L(R) \subseteq L(S)\}$. Prove that A is decidable.

If $L(R) \subseteq L(S)$ then $L(R)$ is completely contained within $L(S)$. This means $L(R) \cap \overline{L(S)} = \emptyset$. Define a new language $L(T) = L(R) \cap \overline{L(S)}$. Let TM N be the TM that decides E_{DFA} from theorem 4.4 in Sipser.

Build a TM M as follows:

M: "On input $\langle R, S\rangle$ where R and S are regular expressions:

1. Create DFA D_T for $L(T)$ as described. [$L(T) = L(R) \cap \overline{L(S)}$]
2. Run TM N from 4.4 on input $\langle D_T \rangle$.
3. If N accepts, accept. If N rejects, reject."

M now decides $L(R) \cap \overline{L(S)}$ which also decides A. □

 Name: Jake Lopez

5. Mapping Reducibility (10 points)

   Proof the following for every language A:

   A is decidable iff $A \leq_m 0^*1^*$

   Remember that X iff Y indicates *"X if and only if Y"*, that is X implies Y and Y implies X.

If $A \leq_m 0^*1^*$ => A is decidable as 0^*1^* is a regular expression and all regular expressions are decidable. And by Sipser theorem 5.22 If $A \leq_m B$ and B is decidable then A is decidable.

If A is decidable then some TM N decides A. Run input $\langle w \rangle$ on N. If N accepts $\langle w \rangle$ output 00, If N rejects $\langle w \rangle$ output 10. As $00 \in 0^*1^*$ and $10 \notin 0^*1^*$ this is a mapping function $f: \Sigma^* \to \Sigma^*$

$w \in A \iff f(w) \in 0^*1^*$.

This means $A \leq_m 0^*1^*$. □

10. Undecidability (10 points)

A *not used* state in a Turing Machine (TM) is a state that is never enterered during the computation of the TM on any input string. For this question, consider the problem of determining whether in a TM a state is not used.

(a) Formulate this problem as a language (3 points).

$L = \{\langle M, q \rangle$ | M is a tm and q is a state never visited in M for any input w $\}$

(b) Prove that this problem is undecidable (7 points). **Do not use Rice's theorem.**

Assume L is decidable. Then there is a TM R that decides L.

Define a TM S as follows:

S: "On input $\langle M, w \rangle$ where M is a TM and w is a string:

1. Run TM R on input $\langle M, w, q \rangle$ where q is a state in M.
2. If R accepts accept, reject otherwise"

If q_{accept} is fed to the TM with no other states or transitions then it is accepted by R and S. However this is equivalent to $\emptyset$ which would make E_{TM} undecidable. But that is a contradiction $\square$

 Name: Jake Lopez

9. Decidability (10 points)

$B = \{\langle M \rangle \mid M$ is a TM that always halts (no matter what the input it) within 50 steps$\}$. Show that B is decidable.

Let M be a decider for B:

M = " On input ⟨M, w⟩ M is a turing machine and w is a string:

1. Create a TM T that simulates M on input w and encode results as ⟨R⟩.
2. Create a DFA V that accepts on strings of length 50 and rejects everything else.
3. Simulate V on input ⟨R⟩.
4. If V accepts, accept. Else, reject."

The encoding R will either be 50 steps long or not which can be accepted by DFA V.

This M is then a decider for B □

8. Turing Machine (10 points)

   Give an implementation-level description of a Turing machine that decides the following language over the alphabet $\{0, 1\}$.

   $L = \{w \mid w \text{ contains twice as many 0s as 1s}\}$

   Give details on how the TM moves its head and the way it stores data on its tape. Do not give details of states or transition functions.

M = " On input <w> :

1. Tape head moves left to # marker.
2. tape head moves right marking each 1.
3. When the tape head reaches end of tape move left until a # is read.
4. move right until a marked 1 is read. and replace with R.
   4.1. move right and replace two zero's with X. If the tape end is reached before two zeros are marked reject.
   4.2. Move left until R is read and repeat.
5. When all marked one's are read read entire tape from left to right.
6. If anything other than R or X is on tape reject. Otherwise Halt. "

7. Decidability (10 points)

Let $A = \{\langle M \rangle \mid M$ is a DFA which accepts every string containing an even number of 1s$\}$. Show that A is decidable. In your proof, you can use any of the theorems from Sipser's book.

Let T be a TM as follows:

T = "On input $\langle N \rangle$ where N is a DFA:

1. Build a new DFA D that accepts the language of strings not containing even number of 1's $\cap$ A.
2. Run TM from Sipser 4.4 on $\langle D \rangle$.
3. If TM from 2 accepts, accept. If TM rejects reject."

Since Language defined in 1 $\cap$ A $= \emptyset$ if there are no even number of 1's this TM T decides A. $\square$

6. Turing Machines (10 points)

A Turing machine with doubly infinite tape is similar to an ordinary Turing machine, but its tape is infinite to the left as well as to the right. The tape is initially filled with blanks except for the portion that contains the input. Computation is defined as usual except that the head never encounters an end to the tape as it moves leftward. Show that this type of Turing machine recognizes the class of Turing-recognizable languages

A TM with doubly infinite tape can simulate a regular TM by choosing an arbitrary point and marking it so that should the tape head reach that posistion it cannot move the tape to the left any more.

For a ordinary TM to simulate a doubly infinite tape TM it must choose an arbitrary point on the tape and divide it into two halves. The TM must then remember the location of the cut so that it can move the tape head to the correct locations as the doubly infinite TM.