SOLVED MIDTERM PAPERS

CS701-Theory of Computation

Presented by Muhammad Sadaqat Ali

Email: muhammadsadaqatali@yahoo.

Question No 1:- Show that the Post Correspondence Problem (PCP) is decidable over the unary alphabet.

Answer:- The PCP over a unary alphabet is decidable. we describe a TM M that decides unary PCP.

Given a unary PCP instance

$$\left\{ \left[\frac{1^{a_1}}{1^{b_1}}\right], \ldots\ldots, \left[\frac{1^{a_n}}{1^{b_n}}\right] \right\}$$

All material available at mscsvu.blogspot.com: Books, Past Assignments, Mid and Final Term Papers

http://mscsvu.blogspot.com/2015/06/recommended-for-those-whose-are-doing.html

Email: muhammadsadaqatali@yahoo.com

M = "On input $\langle a_1, b_1, \ldots\ldots, a_n, b_n \rangle$:

1. check if $a_i = b_i$ for some i. If so, accept.
2. check if there exist i, j such that $a_i > b_i$ and $a_j < b_j$. If so, accept. otherwise, reject."

Question No. 2:

prove that every $t(n)$-time k-tape TM has an equivalent $O(t^2(n))$-time single tape TM.

Answer: Given a k-tape TM M, we can made 1-tape TM N. N works as follows:-

① On input x, Convert input x to $\# q_0 \# x \# \dots \#$ (the start configuration of M). This configuration says that x is on the first tape. The rest of the tapes are empty and the machine is in q_0.

② In each pass over the tape, change the current configuration to the next one.

③ If an accepting configuration is reached, Accept.

④ If an rejecting configuration is reached, Reject.

Now, we have to estimate, how much time does N require.

On any input x of length n, we make the following claims:-

① M uses at most $t(n)$ cells of its k-tapes.

2. Each configuration has length at most $k\,t(n) = O(t(n))$
3. Each pass of N requires at most $O(t(n))$ steps.
4. N makes at most $t(n)$ passes on its tape.

This shows that N runs in time

$$O(t(n) \times t(n)) = O(t^2(n)).$$

Here, we use the fact $t(n) \geq n$.

Thus, the machine converts x to the initial configuration. This takes time $O(n)$.

Thus, the total time is

$$O(n) + O(t^2(n)) = O(t^2(n)).$$

Question No 3:-

Show that A_{TM} is not mapping reducible to E_{TM}

All material available at
mscsvu.blogspot.com:
Books, Past Assignments, Mid and Final
Term Papers

Answer:-

Recall that A_{TM} is undecidable, but it is recognizable so its complement $\overline{A_{TM}}$ is not recognizable.

Note that $\overline{E_{TM}}$, the complement of E_{TM} is recognizable, and so since we know E_{TM} is undecidable, it is also not recognizable.

Proof:- we give a proof by contradiction.

Assume, it is false that A_{TM} is NOT mapping reducible to E_{TM}, so $A_{TM} \leq_m E_{TM}$.

Consider $\overline{A_{TM}}$. Since A_{TM} is mapping reducible to E_{TM}, we immediately get $\overline{A_{TM}}$ is mapping reducible to $\overline{E_{TM}}$.

Since, $\overline{E_{TM}}$ is recognizable, $\overline{A_{TM}}$ is also recognizable. But this is false, hence a contradiction.

Question No 4:-

Which of the following pairs of numbers are relatively prime? Show the calculations that led to your conclusions. a. 1274 & 10505 b. 7289 and 8029

Answer:-

a) We can use the Euclidean algorithm to find the greatest common divisor.

$$10505 = 1274 \times 8 + 313$$
$$1274 = 313 \times 4 + 22$$
$$313 = 22 \times 14 + 5$$
$$22 = 5 \times 4 + 2$$
$$5 = 2 \times 2 + 1$$
$$2 = 1 \times 2 + 0$$

The greatest common divisor of 10505 and 1274 is 1. Therefore, they are relatively prime.

b)

$$8029 = 7289 \times 1 + 740$$
$$7289 = 740 \times 9 + 629$$
$$740 = 629 \times 1 + 111$$
$$629 = 111 \times 5 + 74$$
$$111 = 74 \times 1 + 37$$
$$74 = 37 \times 2 + 0.$$

The greatest common divisor of 8029 and 7289 is 37. Therefore, they are not relatively prime.

Question No. 5:-

G is a digraph and show that PATH is in Class P.

OR.

Design a polynomial time algorithm that takes as input a graph G and two vertices s and t and decides if there is a path from s to t.

Answer:-

we have to give a polynomial time algorithm for this problem. That is

"start BFS or DFS from s and if t appears then there is a path from s to t."

Algorithm

① on input <G,s,t> where G is a digraph, Mark s.

② Repeat till no additional nodes are marked.

③ Scan the edges of G. If an edge (a,b) is found going from a, marked node a to an unmarked node b, mark b.

④ If t is marked, accept. otherwise, reject.

Now, we have to compute the size of input. we know that input size is at least m, where m is the number of nodes in G. Thus, we have to show that algorithm runs in time polynomial in m.

The repeat loop can at most run for m time. Each time all the edges are scanned. Since the number of edges is at most m^2, thus step 2 takes at most m^2 time.

All material available at
mscsvu.blogspot.com:
Books, Past Assignments, Mid and
Final Term Papers

So, the total time is at most

$$m^3.$$

Hence, we have shown that PATH is in P.

Question No. 6:- A Turing machine with ~~say~~ stay put instead of left is similar to an ordinary Turing machine, but the transition function has the form

$$\delta : Q \times T \to Q \times T \times \{R, S\}$$

At each point the machine can move its head right or ~~left~~ let it stay in the same position.

Show that this Turing machine variant is not equivalent to the usual version. What class of languages do these machine recognize?

<u>Answer:-</u> Remembering what it has written on tape cells to the left of the current head position is unnecessary, because the TM is unable to return to these cells an read them.

Using NFA in the actual construction is convenient because it allows ϵ moves which are useful for simulating the "stay put" TM transitions.

The transition function δ' for the NFA is constructed according δ.

- First, we set $\delta'(q_{start}, !) = (q_{0p})$, where q_0 is the start state of TM variant.
- Next, we set $\delta'(q_{accept}, i) = (q_{accept})$ for any i.
- If $\delta(P, a) = (q_{accept}, b, w)$, where $w = R$ or S, we set $\delta'(q_{pa}, \epsilon) = \{q_{accept}\}$.

- If $\delta(P, a) = (q_{reject}, b, w)$, where $w = R$ or S, we set $\delta'(q_{pa}, \epsilon) = \{q_{reject}\}$.

Question No 7:. Show that the collection of Turing-recognizable languages is closed under the operation of

(i) union  (ii) concatenation

Answer:-

(i) For any two Turing-recognizable languages L_1 and L_2, let M_1 and M_2 be the TMs that recognize them. we construct a TM M' that recognizes the union of L_1 and L_2:

" On input w:

1. Run M_1 and M_2 alternatively on w step by step. if either accept, accept. If both halt and reject, then reject."

If any of M_1 and M_2 accept w, M' will accept w since the accepting TM will come to its accepting state after a finite number of steps. Note that if both M_1 and M_2 reject and either of them does so by looping, then M will loop.

(ii) For any two Turing-recognizable languages L_1 and L_2, let M_1 and M_2 be the TMs that recognize them. We construct a NTM M' that recognizes the concatenation of L_1 and L_2:

"On input w:

1. Nondeterministically cut w into two parts $w = w_1 w_2$.
2. Run M_1 on w_1. If it halts and rejects, reject. If it accepts, go to stage 3.
3. Run M_2 on w_2. If it accepts, accept. If it halts and rejects, reject."

If there is a way to cut w into two substrings such M_1 accepts the first part and M_2 accepts the second part, w belongs to the concatenation of L_1 and L_2 and M' will accept w after a finite number of steps.

Question No. 8:- In the silly Post Correspondence Problem, SPCP, in each pair the top string has the same length as the bottom string. Show that SPCP is decidable.

Answer:- The SPCP problem is decidable. It follows from the following claim:-

Claim: A given SPCP instance has match if and only if there is a domino $\left[\frac{t_i}{b_i}\right]$ such that $t_i = b_i$.

<u>Proof of the claim:-</u>

"$\Rightarrow$": If a SPCP instance has a match, it has to start with some domino. Because the length of the top and the bottom string is the same in all dominos, the first domino in the match must surely have the same top and bottom string.

"$\Leftarrow$": If there is a domino with the same top and bottom string then this single domino forms a trivial match

of SPCP.

Finally, checking whether there is a domino with the same top and bottom string is easily decidable by examining the SPCP instance.

Question No. 9:- A useless state in a Turing machine is one that is never entered on any input string. Consider the problem of determining whether a Turing machine has any useless states. Formulate this problem as a language and show that it is undecidable.

Answer:-

Let $USELESS_{TM} = \{\langle M \rangle \mid M$ is a TM with 1 or more useless states. we show $USELESS_{TM}$ is undecidable by reducing E_{TM} to it.

Let R be a TM that decides $USELESS_{TM}$ and construct a TM S that decides E_{TM} as follows:-

S = " On input $\langle M, w \rangle$:

① First we construct a TM N that simulates M on w but ensuring that all of M's original states are not useless. The TM N has input alphabet $\{0, 1, 2\}$. On input 0, N goes through each of the original states of M, except for q_{accept} and q_{reject}, until all nonhalting states of M have been used, and then enters state q_{accept}.

On input 1, N enters state q_{reject}.

On input 2, N simulates M on w. Finally, N restores the original symbol to where the new symbol had been written, and branches to M's start state state to simulate M. If that simulation is about to enter M's accept state, it instead enters a new state r.

② Run $USELESS_{TM}$ on input $\langle N \rangle$. If it accepts, reject. if it reject, accept."

The only possible useless state of N is the state r, because on input 0 and 1, it uses all other states. The only way for N to use state r, is if M accepts w. So, testing whether N has any useless states, in effect tests whether M accepts w.

Note that the hardest part of this problem is making sure that the machine N has no "unexpected" useless states.

Question No 10:- Let B be the set of all ~~inif~~ infinite sequence over $\{0,1\}$. Show that B is uncountable, using a proof by diagonalization.

Answer:- Suppose B is countable and a correspondence $f: N \rightarrow B$ exists. We construct x in B that is not paired with anything is N.

Let $x = x_1 x_2 \ldots$. Let $x_i = 0$ if $f(i)_i = 1$, and $x_i = 1$ if $f(i)_i = 0$ where $f(i)_i$ is the ith bit of $f(i)$. Therefore, we ensure that x is not $f(i)$ for any i because it differs from $f(i)$ in the ith symbol, and a contradiction occurs.

Question No. 11:- Let X be the set $\{1, 2, 3, 4, 5\}$ and Y be the set $\{6, 7, 8, 9, 10\}$. We describe the function $f: X \rightarrow Y$ and $g: X \rightarrow Y$ in the following tables. Answer each part and give a reason for each negative answer.

n	$f(n)$
1	6
2	7
3	6
4	7
5	6

n	$g(n)$
1	10
2	9
3	8
4	7
5	6

a. Is f one-to-one?
b. Is f onto?
c. Is f a correspondence?
d. Is g one-to-one?
e. Is g onto?
f. Is g a correspondence?

Answer:-

a. g is one-to-one. f is not one-to-one because $f(1) = f(3)$

b. g is onto. f is not onto because there doesnot exist $x \in X$ such that $f(x) = 10$.

c. g is a correspondence because g is one-to-one and onto. f is not a correspondence because f is not one-to-one and onto.

Question No. 12:- Let $T = \{(i, j, k) \mid i, j, k \in N\}$. Show that T is countable.

Answer:- We demonstrate a one-to-one $f: T \to N$. Let $f(i, j, k) = 2^i 3^j 5^k$. Function f is one-to-one because if $a \neq b$, $f(a) \neq f(b)$. Therefore, T is countable.

Question NO 13:- Choice the correct one:-

1. EMPTINESS problem for LBA is decidable?
   - a. TRUE
   - b. FALSE ✓

2. A property that holds for almost all strings also holds for incompressible strings.
   - a. TRUE
   - b. FALSE ✓

3. A string x is b-compressible if $K(x) > |x| - b$?

a. TRUE ✓
b. FALSE

4. Turing machines accept/reject their own description.

a. TRUE ✓
b. FALSE

5. A correspondence or one-to-one correspondence is a function that is both one to one and onto.

a. TRUE ✓
b. FALSE

6. Which string belongs to L such that $i+j=k$?

a. aabb
b. aabbcccc ✓
c. aaabbcc
d. aabbbcc

Question No. 14:- Show that MPCP is undecidable.

Answer:- Assume that MPCP is decidable. Let us say, we have a decider R for MPCP. Consider the following decider S

1. On input <M, w>
2. Construct P' as described in the seven parts.
3. Run R on P'.
4. If R accepts, accept.
5. if R rejects, reject.

Then S is a decider for A_{TM}. Which is a contradiction to the fact that A_{TM} is undecidable.

Question No. 15:- choose the correct one:-

1. Can we design a 1-tape TM that accept A in $o(n \log n)$ time. Preferably linear time?
   a. YES
   b. NO ✓

2. The machine has seven possible ways to proceed.

$$\delta(q_5, a) = \{(q_7, b, R), (q_4, c, L), (q_3, a, R)\}$$

   a. YES
   b. NO ✓

3. A string is b-incompressible if it is not b-compressible.
   a. YES ✓
   b. NO

Question No. 16:- Write 7 parts of construction PCP?

Answer:- part 1:- put $\left[\frac{\#}{\# q_0 w_1 w_2 \cdots w_n \#}\right]$ in P' as the first domino.

part 2:- For every $a, b \in \Gamma$ and $q, r \in Q$ where $q \neq q_r$ if

$$\delta(q, a) = (r, b, R) \quad \text{put } \left[\frac{qa}{br}\right] \text{ in } P'.$$

part 3:- For every $a, b, c \in \Gamma$ and $q, r \in Q$ where $q \neq q_r$ if

$$\delta(q, a) = (r, b, L) \quad \text{put } \left[\frac{cqa}{rcb}\right] \text{ in } P'.$$

part 4:- For every $a \in \Gamma$ put $\left[\frac{a}{a}\right]$ in P'.

part 5:- put $\left[\frac{\#}{\#}\right]$ and $\left[\frac{\#}{\sqcup \#}\right]$.

part 6:- For every $a \in \Gamma$ put $\left[\frac{a q_a}{q_a}\right]$ and $\left[\frac{q_a a}{q_a}\right]$ in P'.

part 7:- Add $\left[\frac{q_a \#\#}{\#}\right]$ in P'.

which will allow us to complete the match.

Question No. 17:- If $A \leq_m B$ and B is Turing recognizable then A is Turing recognizable.

Answer:- Let $A \leq_m B$ and let f be the reducibility from A to B. Furthermore, since B is Turing recognizable, there is a TM M such that

$$L(M) = B$$

Consider N:

"On input x

① Compute $y = f(x)$

② Run M on y

③ if M accepts, accept.

then it is easy to see that $L(N) = A$.

ALLAH HAFIZ

Important Links:

http://mscsvu1.blogspot.com/p/guidance.html

http://mscsvu.blogspot.com/2015/06/cs-702-handouts-01-22.html http://mscsvu.blogspot.com/2015/06/guidelines-and-requirements-for-writing.html

http://mscsvu.blogspot.com/2015/06/writing-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs713-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs710-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs706-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs702-mid-term-papers_25.html

http://mscsvu.blogspot.com/2015/06/cs712-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs701-mid-term-papers_25.html

http://mscsvu.blogspot.com/2015/06/cs703-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs708-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/cs708-mid-term-papers.html

http://mscsvu.blogspot.com/2015/06/mid-term-past-papers.html

http://mscsvu.blogspot.com/2015/06/midterm-paper-cs-703-operating-system.html

http://mscsvu.blogspot.com/2015/06/recommended-for-those-whose-are-doing.html

http://mscsvu.blogspot.com/2015/06/cs-701-handouts-01-22.html

http://mscsvu.blogspot.com/2015/06/general-discussion-forum.html

http://mscsvu1.blogspot.com/p/guidance.html

http://mscsvu.blogspot.com/2015/06/academic-research-paper-thesis-forum.html

http://mscsvu.blogspot.com/2015/06/software-engineering-courses-forum.html

http://mscsvu.blogspot.com/2015/06/database-courses-forum.html

http://mscsvu.blogspot.com/2015/06/networking-courses-forum.html

http://mscsvu.blogspot.com/2015/06/core-courses-forum.html

HERE ARE MEGA COLLECTION OF BOOKS: http://mscsvu.blogspot.com/p/1.html

http://mscsvu.blogspot.com/p/cs721.html

http://mscsvu.blogspot.com/p/cs717.html

http://mscsvu.blogspot.com/p/cs713.html

http://mscsvu.blogspot.com/p/cs712.html

http://mscsvu.blogspot.com/p/cs711.html

http://mscsvu.blogspot.com/p/cs724-software-process-improvement.html

http://mscsvu.blogspot.com/p/cs726.html

http://mscsvu.blogspot.com/p/cs709.html

http://mscsvu.blogspot.com/p/cs706.html

http://mscsvu.blogspot.com/p/cs704-ad.html

http://mscsvu.blogspot.com/p/cs703.html

http://mscsvu.blogspot.com/p/cs702-advanced.html

http://mscsvu.blogspot.com/p/cs701.html

http://mscsvu.blogspot.com/2015/06/term-paper-formats-ieee-format-for-mscs.html

CS708: http://mscsvu.blogspot.com/p/blog-page_8.html

CS710: http://mscsvu.blogspot.com/p/blog-page_88.html

CS720: http://mscsvu.blogspot.com/p/blog-page_92.html

CS723: http://mscsvu.blogspot.com/p/blog-page_18.html