7

[Exam 1] Name: Karimdad Niaz Khaskheli ID: 05088

Fall 2023: CS 313: Computational Complexity Theory

Due: 12:45 pm, Thursday, October 5, 2023. Total Marks: 24

This exam copy contains 3 pages, including this one.

Question 1 4.5/[12 points]

For each part, provide brief explanations and/or proofs.

1. We have defined a relation $\leq_p$ among languages. This relation is reflexive (i.e. $L \leq_p L'$ for all languages) and transitive (i.e. if $L \leq_p L'$ and $L' \leq_p L''$ then $L \leq_p L''$). Why is it not symmetric, namely, why is it that $L \leq_p L'$ need not imply $L' \leq_p L$?

The languages may differ entirely and since L' is $\{0,1\}^* \in L$, the relation may not always backtrack.

invalid ?

C X

2. Show that **NP** is closed under concatenation.

?

$L_1 \in NP$ & $L_2 \in NP$

This means that for an $x \in \{L_1 \circ L_2\}$, it can at most have a U of length $p(n)$ which would be ~~$(|L_1| \circ |L_2|)$~~ ~~$(|L_1| \times |L_2|)$~~ $(|L_1| \times |L_2|)$.

This length will still be polynomial, ~~however since the size~~ for any n_1 and n_2 sizes for L_1 and L_2 respectively.

X ~~It~~ should be in terms of lengths ~~length~~ cardinality of the languages?

3. Why is every **NP-Hard** language not decidable by a Turing Machine?

NP-Hard Languages imply that the problem is atleast as difficult as the most difficult problems in NP. There may exist an NP-Hard problem that is not in NP and for which a TM never halts and loop indefinitely. In addition there is a limitation on the amount of resources a TM can use to decide a problem, if there is a problem that demands more than what is available ~~TM~~ the TM will fail to decide it.

↳ this qs is about undecidable languages, so amount of resource is not an issue.

which problem?

4. Show that if **P** = **NP** then **NP** ⊂ **EXP**, where ⊂ denotes the proper subset relation.

If P=NP then this means that the problem has an efficient solution that can be verified in Polytime. Since EXP is the class of problems that can be solved by a DTM in exponential time, then there must exist a naive solution for all problems in NP (when P=NP).

LA

But why is NP ⊂ EXP?

vague

If NP=P, then ∀ problem in NP can be solved in d polytime. Exp... that some problem cannot be solved in polytime, hence NP ⊂ EXP

1.5/

Question 2 [6 points]

Show that the language

VCF = { (G, k, F) | Undirected graph G has a vertex cover[1] of size k, such that at least one of the vertices from each pair of vertices in the forbidden set[2] F is not in the vertex cover }.

is NP-Complete.

It is not possible to simply obtain the vertices in polytime.

The vertex cover would make the set of vertices in that span the entire graph.

make the problem P

VCF is in NP as we can run a graph traversal algorithm on the graph for length 'k' and check whether the requirements for VC are met. Once ~~the graph~~ a set of vertices for VC is obtained we ~~If such a vertex exists that~~ then check it against the pairs in F. If the latter condition * is met we accept else we reject. This is computed in polynomial time and hence VCF is in NP. * given in the language definition for VCF.

X

VCF $\leq_p$ VC ∴ since we know 3SAT is NP-Complete and VC $\leq_p$ 3SAT

other way round.

certificate u will be the vertex cover in VCF. ~~Since we know that~~ If a vertex cover of size k satisfies VCF for a graph G, it automatically implies that it is a valid vertex cover for VC since it is free from the forbidden set restriction. Since we know that VC $\leq_p$ 3SAT, and VCF $\leq_p$ VC, we can safely assert that VCF $\leq_p$ 3SAT using transitivity.

Hence VCF is NP-complete.

need to show

3SAT $\leq_p$ VCF

correct idea but incorrectly applied

[1] a set of vertices that includes at least one vertex from every edge of the graph ✓

[2] the forbidden set contains pairs of vertices ✓

forbidden edges i think

1.5/3

Question 3 1/ [6 points]

Show that **NP** = **coNP** *iff* **3SAT** and **TAUTOLOGY** are polynomial time reducible to one another.

This means that ~~there exists~~ if there exists a valid reduction for 3SAT to TAUT and vice versa it would imply that NP = coNP. that is one side of iff.

since 3SAT is a known NP-Complete Problem. We can show that TAUT has a reduction to it. and that $\overline{3SAT}$ has a reduction to TAUT

For a language $L \subseteq \{0,1\}^*$, ~~if there exists~~ and an $u \in \{0,1\}^*$ and a TM $M(x,u)$ for 3SAT that decides whether u ~~decides~~ is ~~valid or~~ accepted or not,

~~$x \in L$~~

$x \in L \iff \exists u \in \{0,1\}^{p(|u|)}, M(x,u) = 1$ is this meant to imply complement?

Then for an $x \in \{0,1\}^*/L$ i.e. $x \in L'$, there exists a $f(x)$ for which $M'(f(x))$ ~~decides~~ (for $\overline{TAUT}$), decides it.

~~3SAT are in~~ TAUT is in NP as we can find a valid ~~valid~~ assignment to the variables for which ~~the truth value of the ... is ∅ is~~ if ∅ is satisfied in polynomial time.

The certificate for ~~this~~ $\overline{TAUT}$ will be the assignment of values to the variables such that ∅ is satisfied.

~~[illegible]~~ $3SAT \leq_p \overline{TAUT}$

$TAUT \leq_p \overline{3SAT}$

This is true by definition of $3SAT \in NP$-Comp, $TAUT \in coNP$-Comp

This entails that we need to find an assignment in L_{TAUT} which can be modified into an assignment for $L'_{\overline{3SAT}}$ such that ~~∅~~ $\emptyset \in \overline{3SAT}$ is satisfied, (A; $x \in L$ $M_{TAUT}(x,u) = 1$, $M'_{\overline{3SAT}}(f(x),u) = 1$

~~Since, every NP-complete has a Complement [illegible] coNP-complete problem and as shown above $3SAT \leq_p \overline{TAUT}$ which is a complement to TAUT~~

~~3SAT / $\overline{TAUT}$ 3SAT / TAUT~~

CamScanner

For an ~~$x \in$~~ L ($L \subseteq \{0,1\}^*$, $x \in \{0,1\}^*$

if ~~$x \Longleftrightarrow L$~~ $x \in L \iff \forall u \in \{0,1\}^{p(|x|)}$, $M(x,u)=1$ | $\emptyset$ is a tautology;

since TAUT is coNP-complete there must exist a ~~to~~ $f(x)$ ~~that map~~ ~~$x \in L$~~ for which a $M'(f(x))=1$, & that is, which satisfies $\overline{TAUT}$ since $f(x) \in L'$ $\therefore$ $L' \in \{0,1\}^* - L$.

$\boxed{\text{since } 3SAT \le_p \overline{TAUT} \text{ (as shown earlier) and } \overline{TAUT} \le_p TAUT}$

can't be true unless $NP = coNP$.

$NP = coNP$ ~~iff~~

There are errors of notation in the proof that make it difficult to follow it.

You can explain the proof to me during office hours.

X