1

(Sipser 9.12) Describe the error in the following "proof" that $P \neq NP$. Assume that $P = NP$, then $SAT \in P$ and so for some $k, SAT \in TIME(n^k)$. Because every language in NP is polynomial time reducible to SAT, you have $NP \subset TIME(n^k)$. By the time hierarchy theorem, $TIME(n^{k+1})$ contains a language that is not in $TIME(n^k)$, which contradicts $P \subset TIME(n^k)$. Therefore $P \neq NP$

> The error in this proof comes in the line that "every language in NP is polynomial time rediculble to SAT, then $NP \subset TIME(n^k)$". While it is true that every language is in NP is polynomial time reducible to SAT, it is not guarenteed that the function f that reduces L to SAT is less than n^k. The correct conclusion to their sentence is $NP \subset TIME(O(f(n)+n^k))$. This then makes their following argument incorrect because we cannot determine if $TIME(O(f(n) + n^k)) \subset TIME(n^{k+1})$ because $f(n)$ is not a constant term for all languages in NP.

2

Prove that $USAT \in P^{SAT}$

3

Prove that an oracle C exists for which $NP^C \neq coNP^C$

> Suppose we have language $L = \{w | \exists x x \in C \wedge |x| = |w|\}$
>
> The compliment of this language is $\bar{L} = \{w | \forall x x \notin C \vee |x| \neq |w|\}$
>
> Let M_L be a NTM that decides L, we need to prove that M_L halts in polynomial time.

4

Prove that in an interactive proof, if the verifier is required to be a deterministic, polynomial time algorithm with no access to random bits, then the class of languages this system can decide is equal to NP, even if we allow an arbitrary number of queries to the prover.