Homework 9

Math 167 / CS 142: Complexity Theory

1. Problem 14.5.10 of the text. (Hint: Do part (b) first.)

Define a *robust* oracle machine $M^?$ deciding language L to be one such that $L(M^A) = L$ for all oracles A. That is, the answers are always correct, independently of the oracle (although the number of steps may vary from oracle to oracle). If furthermore M^A works in polynomial time, we say that oracle A helps the robust machine $M^?$. Let \mathbf{P}_h be the class of languages decidable in polynomial time by deterministic robust oracle machines that can be helped; and \mathbf{NP}_h for nondeterministic machines.

- (a) Show that $P_h = NP \cap coNP$.
- (b) Show that $\mathbf{NP}_{\mathrm{h}} = \mathbf{NP}.$
- (a)
- (b)
- 2. Reduce the computation of the parity $x_1 \oplus \cdots \oplus x_n$ of n Boolean variables to the multiplication of two n^2 -bit factors (represented in binary) to form a $2n^2$ -bit product, by setting some of the bits of the factors to constants in such a way that the parity of the remaining bits emerges as one of the bits of the product.
- 3. Show how the n Boolean functions $y_1=x_1,y_2=x_1\vee x_2,\ldots,y_n=x_1\vee x_2\vee\cdots\vee x_n$ of the n Boolean variables x_1,x_2,\ldots,x_n can be computed by an unbounded fan-incircuit of NOT-, AND- and OR-gates of size $\mathcal{O}(n)$ and depth $\mathcal{O}(1)$. (Hint: Partition the n inputs into about \sqrt{n} blocks of about \sqrt{n} inputs each.)