The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual
Chapter 6, Problem 6.8

What we need to prove is that, if some problem A is in NP and some problem B many-one reduces to A in polynomial time, then B is also in NP. Since A is in NP, there is a Turing machine M and a polynomial p such that a certificate for a "yes" instance x of A can be verified by M in time $p(|x|)$. Let f be the polynomial-time transformation from B to A and let q be its polynomial bound. We build a checker M' for B by using the checker M for A and the the transformation f. Specifically, M' will expect certificates for instances of A, not B, because it will verify such a certificate for an instance x of B by transforming x into an instance $y = f(x)$ of A in time $q(|x|)$, running M on y in time $p(|y|)$, and returning the answer given by M. Since the transformation takes time $q(|x|)$, what it produces, the string y, cannot exceed $q(|x|)$ in length; thus the time taken by checker M, which is $p(|y|)$, is at most $p(q(|x|))$, which is a polynomial in $|x|$. Hence our checker M' runs in time polynomial in $|x|$ and thus B is in NP.