The Theory of Computation
Bernard M.E. Moret
Addison-Wesley-Longman, 1998

Solution Manual
Chapter 7, Additional Problem 2

That the problem is in NP is immediate: *Satisfiability* is in NP and we need only make one additional check to verify that each clause either is a Horn clause or has two literals. We prove that the problem is NP-complete by reducing *Satisfiability* to it.

If a clause of *Satisfiability* has one or two literals, we pass it unchanged; *ditto* if it is a Horn clause. We provide an "alter ego" for (the complement of) each of our variables as follows: for variable x, we create the new variable X and add the two clauses $\{\overline{x}, \overline{X}\}$ and $\{x, X\}$. Together, these two clauses enforce $X \equiv \overline{x}$. Now, given some clause of *Satisfiability* with $m > 2$ literals and $k > 1$ of them complemented, we replace every complemented literal in the clause by its alter ego and place the result in the new collection of clauses.

The resulting collection of clauses is logically equivalent to the original collection and all of the new clauses are Horn clauses or have length two. Given an instance of *Satisfiability* with n variables and m clauses, we produce a new instance with $2n$ variables and $m + 2n$ clauses, in a process that obviously runs in polynomial time.