# The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

## *Solution Manual*

## *Chapter 6, Additional Problem 6.3*

One direction is easy. If a language is regular, it is accepted by a DFA; the DFA can be viewed as a highly restricted off-line Turing machine that uses no space at all on its scratch tape and so certainly runs in constant space.

Reducing constant space to zero space in the other direction is also easy: if our off-line Turing machine uses space $k$, we can produce a new off-line Turing machine with a control state set that encodes the $|\Sigma|^k$ possible configurations of the scratch tape of our original machine—in effect, we simply multiply the number of states by $|\Sigma|^k$, a constant. The problem is that the resulting machine is still a Turing machine and can do something that no FA can do, namely go back and reread some input it has already seen. What we have is, in effect, a *two-way finite automaton*.

So the trick is to show that whatever such a machine accepts can also be accepted by a regular FA. This result was proved as early as 1959 by J.C. Shepherdson, "The reduction of two-way automata to one-way automata," *IBM J. Res. & Dev.* **3**, pp. 198–200. Since the reduction is rather tricky and since the original paper is quite clear and concise, we simply refer the reader to the original write-up.