Summary: We propose clean reversible simulations of ranking binary trees and unranking as reversible algorithms for reversible computing systems, which are useful for enumerating and randomly generating binary trees. Algorithms for ranking binary trees and their inverses have been studied since the 1970s. Each of these algorithms can be converted into a reversible simulation by saving all data and control information otherwise lost, and each pair of ranking and unranking reversible programs can be combined to realize a clean reversible simulation by using the Bennett method. However, such a clean reversible simulation requires multiple traversal of the given data and/or intermediate data as well as additional storage proportional to the length of the computation. We show that for Knott’s ranking and unranking algorithms [G. D. Knott, Commun. ACM 20, 113–115 (1977; Zbl 0345.68025)], additional storage usage can be reduced by using the proper assertions of reversible loops in embedded reversible simulations. We also show a clean reversible simulation that involves only one traversal. The running time and memory usage of the proposed clean reversible simulations are asymptotically equivalent to those of the original programs by Knott with intermediate garbage of constant size. In general, the derivation strategy of efficient reversible programs from irreversible ones has not yet been established, and this study can be seen as one of the case studies. All the reversible programs presented in this paper can be run on an interpreter of the reversible programming language Janus.

For the entire collection see [Zbl 1396.68007].

MSC:
68W40 Analysis of algorithms
68N19 Other programming paradigms (object-oriented, sequential, concurrent, automatic, etc.)
68P05 Data structures

Software:
Janus

Full Text: DOI

References:


[40] Yokoyama, T. · Zbl 1407.68075 · doi:10.1007/978-3-642-29517-1_2

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.