Reducing program size has become an important goal in the design of modern embedded systems targeted to mass production. This problem has driven efforts aimed at designing processors with shorter instruction formats (e.g., ARM Thumb and MIPS16) or able to execute compressed code (e.g., IBM PowerPC 405), This paper proposes three code compression algorithms for embedded RISC architectures. In all algorithms, the encoded symbols are extracted from program expression trees. The algorithms differ on the granularity of the encoded symbol, which are selected from whole trees, parts of trees, or single instructions. Dictionary-based decompression engines are proposed for each compression algorithm. Experimental results, based on SPEC CINT95 programs running on the MIPS R4000 processor, reveal an average compression ratio of 53.6% (31.5%) if the area of the decompression engine is (not) considered.