This paper proposes a code compression technique called operand factorization. The central idea of operand factorization is the separation of program expression trees into sequences of tree-patterns (opcodes) and operand patterns (registers and immediates). Using this technique, we show that tree and operand patterns have exponential frequency distributions. A set of experiments were designed to explore this feature. They reveal an average compression ratio of 43% for SPECInt95 programs. A decompression engine is proposed, which assembles tree and operand patterns into uncompressed instruction sequences. An encoding that improves the design of the decompression engine results in a 48% compression ratio. Compression ratio numbers take into consideration an estimate of the decompression engine size.