

# Instructions to interpret the data files

Prasad Kulkarni (kulkarni@eecs.ku.edu)

1. Line 1: *Function Executed: 0/1* indicates if the current function is executed when the benchmark in run using the input data provided with this benchmark.
2. There may be some information associated with each evaluated phase ordering sequence. This information includes:
  - Is the function instance produced *New* or *Old*. A function instance is old if the same instance was produced earlier using a different phase ordering.
  - To detect if the current function is new or old, we need to compare this instance with all previous function instances. To perform this check efficiently we use various hashes. The *CRC* and *fhash* are the hash values for the current instance, and *r\_CRC* and *r\_fhash* are the hash values for a remapped version of this function.
  - The *control flow* of the function instance is maintained.
  - The code size (*rcount*), dynamic instruction count (*Dynamic Count*), and simulator cycles (*Simulator Count*) is provided were applicable.
3. The actual phase ordering sequence is displayed affixed by the keyword *Sequence:*. The compiler optimization phases are numerically represented as presented in Table 1.

No.	Optimization Phase
1	Branch Chaining
3	Remove Useless Jumps
4	Remove Unreachable Code
5	Reverse Branches
6	Basic Block Reordering
8	Instruction Selection
10	Evaluation Order Determination
12	Register Assignment
13	Minimize Loop Jumps
14	Dead Assignment Elimination
15	Register Allocation
16	Common Subexpression Elimination
21	Strength Reduction
25	Loop Transformations
27	Code Abstraction
29	Loop Unrolling

Table 1: VPO Optimization Phases and Their Representing Numerals

Please contact us if you need more information.