ParkBench Run and Report Rules

---

1. Introduction

This document includes descriptions of the levels of optimization allowable for the ParkBench programs and lists requirements for reporting results. A description of the ParkBench philosophy can be found in [1].

All results publicly disclosed must adhere to the ParkBench Run and Report Rules. The following are expected:

- Adherence to the criteria for use of compiler directives, source modifications, and compiling system flag selection.
- Availability of full disclosure report.

2. Benchmark Compilation and Measurement

1. Baseline Runs

Optimizations as described below are allowed.

1. Compile and load options

Compiler or loader flags which are supported and documented by the supplier are allowed. These include porting, optimization, and preprocessor invocation.

2. Libraries

Linking to optimized versions of the following libraries is allowed:

- BLACS,
- BLAS,
- LAPACK,
- MPI,
- PVM,
- and ScaLAPACK.

Acceptable use of such libraries is subject to the following rules:

- All libraries used shall be disclosed with the results submission. Each library shall be identified by library name, revision, and source (supplier). Libraries which are not generally available are not permitted unless they are made available by the reporting organization within 6 months.
- Calls to library subroutines should have equivalent functionality to that in the released benchmark code. Code modifications to accommodate various library call formats are not allowed.

2. Optimized Runs

1. Compiler Directives

Source code directives which are supported and documented by the supplier of the compiler may be used in the ParkBench benchmark codes. Language extensions instead of directives (such as PARALLEL DO instead of C$DIR PARALLEL) may also be used.

Examples of acceptable directives include:

- Loop level vectorization enabling directives for systems with vector processors
- Loop parallelism directives for multi-processor systems
- Data distribution directives for distributed memory systems

Compiler directives must not be uniquely applicable to the execution of the ParkBench benchmark codes.

2. **Code modifications**

   Examples of allowable modifications are:

   - Removal of parts of the code which are never executed
   - Avoiding unnecessary executions of parts of the program/code
   - Loop unrolling
   - Loop blocking
   - Changes of the data layout
   - Calls to library subroutines

3. Limitations of Optimization

1. **Optimized assembly modules**

   Substituting any part of the code by optimized assembly modules or modification of compiler generated assembly code or executables is discouraged and should be disclosed in reporting the results.

2. **Code with limited calculation accuracy**

   The calculation should be carried out in full precision (64-bit or the equivalent). However the substitution of algorithms is allowed (see **Exchange of the used mathematical algorithm**).

3. **Exchange of the used mathematical algorithm**

   Any change of algorithms must be fully disclosed and is subject to review by the ParkBench Committee. Passing the verification test is a necessary condition for such an approval. The substituted algorithm must be as robust as the baseline algorithm.

4. **Using the knowledge of the solution**

Any modification of the code or input data sets, which uses the knowledge of the solution or of the verification test, is not permitted.

5. **Code to circumvent the actual computation**

   Any modification of the code to circumvent the actual computation is not permitted.

4. Special requirements for single benchmarks

Additional restrictions for single benchmarks might be specified separately in the Readme files (eg. PSTSWM).

3. Disclosing and Publishing ParkBench Results

All ParkBench benchmark results should conform to the rules defined in this document. Documentation should include full disclosure information (defined in section Configuration Disclosure) and necessary and sufficient information for reproducing the results.

1. Configuration Disclosure

The following paragraphs detail the information constituting a full disclosure.

1. **System Identification**

   - System model name
   - ParkBench version number
   - Test sponsor (name, location)
   - Test date (month, year)

2. **Configuration**

   A description of a necessary and sufficient hardware and software configuration should be supplied. For example, this could include information about the CPU type, memory organization, operating system name and level, and names and versions of compilers and loaders.

3. **System Tuning Options**

   Description of special system tuning, including any special OS parameters set, or changes to standard daemons, etc.

4. **Benchmark Code Tuning options**

        - Portability compiler/load flags used by any benchmark
        - Source code directives used in any benchmark code
        - Performance libraries used in creating any benchmark

2. **ParkBench Problem Sizes**

   Benchmark results are reported per application area, per problem size. Up to five problem sizes are provided for each of the benchmarks.

   Keeping all things the same in the system configuration (hardware and software), the benchmarker may report a table of results for a particular size where the only item varied was the number of processes used by the benchmark code.

3. **Test Results Disclosure**

   The actual test results as produced by the benchmarks programs should be copied to a file. They consist of the elapsed times as measured by the program and verification information. See the specific instructions in the distribution of ParkBench, as to which timers are permitted.

4. **Submitting Results to ParkBench Committee**

   Prior to public dissemination, all ParkBench results and configuration disclosures should be sent to ParkBench via email at parkbench-comments@cs.utk.edu for certification. ParkBench intends to publish results periodically and as soon as possible.

   Submitted results should include the following elements:

    - Compiler Flags/Directives descriptions.
    - Algorithmic modifications:
        - Description of algorithmic changes.
        - Library substitutions.
    - All result files written by the benchmarks.
    - All used binaries may be sent to ParkBench. They will be made available to public in a binary repository to permit easy reproducibility of reported results.

Literature

[1] "Public International Benchmarks for Parallel Computers"

Last Updated: May 8, 1996