AMD **Accelerated** Parallel Processing TECHNOLOGY

SAMPLE

GEMM

1 Overview

1.1 Location

`$<APPSDKSamplesInstallPath>\samples\C++Amp\`

1.2 How to Run

See the *Getting Started* guide for how to build samples. You first must compile the sample.

Use the command line to change to the directory where the executable is located. The sample executable is in `$<APPSDKSamplesInstallPath>\samples\C++Amp\bin\x86\` for 32-bit builds, and `$<APPSDKSamplesInstallPath>\samples\C++Amp\bin\x86_64\` for 64-bit builds.

Type the following command(s).

1. `GEMM`

   This runs the program with the default options: `-i 10 -s 2048`

2. `GEMM -h`

   This prints the help file.

Ensure you have installed Microsoft® Visual Studio® 2012 or higher.

1.3 Command Line Options

Table 1 lists, and briefly describes, the command line options.

Table 1 Command Line Options

Short Form	Long Form	Description
`-h`	`--help`	Show all command options and their respective meaning.
`-q`	`--quiet`	Quiet mode. Suppresses text output.
`-e`	`--verify`	Verify results against reference implementation.
`-t`	`--timing`	Print timing.
`-d`	`--deviceId`	Select deviceId to be used (0 to N-1, where N is the ID of the device to be used).
`-v`	`--version`	AMD APP SDK version string.
`-2`	`--size`	Edge length of square matrices A, B, and C.
`-i`	`--iterations`	Number of times to repeat each algorithm.
`-V`	`--arrayview`	Use array_view instead of array.

2 Introduction

The General Matrix Multiply (GEMM) is a subroutine in the Basic Linear Algebra Subprograms (BLAS). It performs matrix multiplication: the multiplication of two matrices. GEMM is often tuned by high-performance computing (HPC) vendors to run as fast as possible.

The GEMM routine calculates the new value of matrix C based on the matrix-product of matrices A and B, as well as the old value of matrix C.

.C = aAB +ßC

where a and ß values are scalar coefficients.

According to the data type, GEMM has four versions: SGEMM, DGEMM, CGEMM, and ZGEMM for handling float, double, complex, and complex-double data types, respectively. This sample implements SGEMM NN/TN and DGEMM NN/TN. N means normal; T means transpose.

3 References

1. Dongarra, Jack J.; Croz, Jeremy Du; Hammarling, Sven; Duff, Iain S. (1990), "A set of level 3 basic linear algebra subprograms", ACM Transactions on Mathematical Software 16 (1): 1–17, doi:10.1145/77626.79170, ISSN 0098-3500.
2. Golub, Gene H.; Van Loan, Charles F. (1996), Matrix Computations (3rd ed.), Johns Hopkins, ISBN 978-0-8018-5414-9.
3. GSL Team (2007), "§12.1.3 Level 3 GSL BLAS Interface", GNU Scientific Library. Reference Manual, http://www.gnu.org/software/gsl/manual/html_node/Level-3-GSL-BLAS-Interface.html.
4. Goto, Kazushige; van de Geijn, Robert A. (2008), "Anatomy of High-Performance Matrix Multiplication", ACM Transactions on Mathematical Software 34 (3): Article 12, 25 pages, doi:10.1145/1356052.1356053.

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: developer.amd.com/appsdk
Developing: developer.amd.com/
Forum: developer.amd.com/openclforum