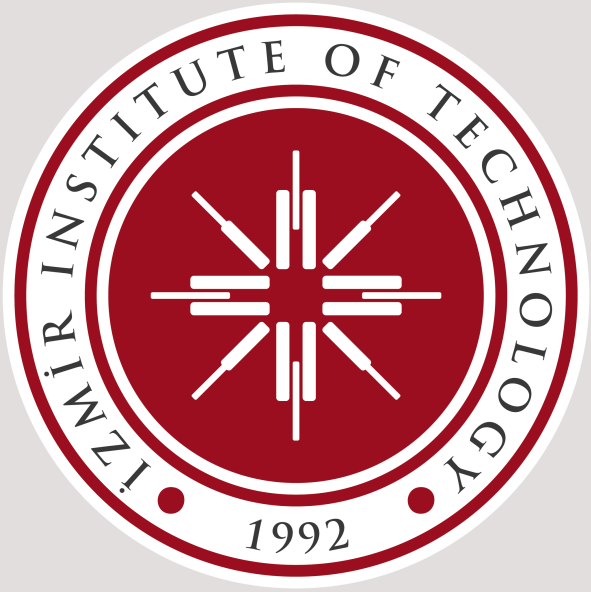
Izmir Institute of Technology
Computer Engineering Department
CENG513 Programming Assignment 1

Student Name: Gökay Gülsoy Student No: 270201072

March 12, 2024



System Configuration

My system configuration for executing the benchmarks is as follows:

- Processor: Intel Core i7-10750H CPU @ 260 GHzX12
- Operating System: Ubuntu 22.04.4 LTS
- Memory: 16,00 GiB
- GCC Compiler: gcc 11.4.0
- Clang Compiler: clang 19.0.0

Building LLVM Test Suite and Running Benchmarks

Commands to be executed for building[1][2][3] and running benchmarks is as follows:

1. Command to check for llvm-lit version: /home/gokay/llvm-project/build/bin/llvm-lit –version
2. Command to clone the llvm-test-suite: git clone https://github.com/llvm/llvm-test-suite.git
3. Command to create build directory: mkdir test-suite-build
4. Command to change directory to build directory: cd test-suite-build
5. Command to write build files into build directory: cmake -D_CMAKE_COMPILER=~ llvm-project/build/bin/clang -C ../cmake/caches/O3.cmake ..
6. Command to make the llvm-test-suite from test-suite-build directory: make
7. Command to run the tests: llvm-lit -v -j 8 -o results.json

I run the 7th command three times for clang and three times for the gcc compiler to obtain execution time benchmarks from Multisource directory. Inorder to build llvm-test-suite with gcc, 5th command can be configured as follows and llvm-test-suite can be rebuild after that with make[4]:

- cmake -D_CMAKE_COMPILER=/usr/bin/gcc -C ../cmake/caches/O3.cmake ..

Following are the applications that I have chosen for running benchmarks:

1. MultiSource/Benchmarks/llubenchmark/llu
2. MultiSource/Benchmarks/7zip/7zip-benchmark
3. MultiSource/Benchmarks/Reductions-flt/Reductions-flt
4. MultiSource/Benchmarks/NPB-serial/is/is
5. MultiSource/Benchmarks/Bullet /bullet

Execution time statistic that I have obtained are as follows:

Benchmark Results GCC				
llu	7zip-benchmark	Reduction-flt	is	bullet
15.5266s	9.2371s	5.8437s	5.7246s	4.2390s
11.6043s	8.6455s	5.2043s	5.3806s	3.4759s
9.9513s	8.5745s	5.1098s	5.8200s	2.8970s

Table 1: Execution time statistics for gcc

Table 1 shows the execution time statistics for the applications that I have tested with gcc. Following table shows execution time statistic for clang:

Benchmark Results Clang				
llu	7zip-benchmark	Reduction-flt	is	bullet
7.8791s	7.3820s	3.8842s	3.2795s	4.4335s
8.5101s	11.8736s	3.9758s	5.3109s	5.0810s
10.9066s	9.4378s	8.9971s	6.2024s	7.0257s

Table 2: Execution time statistics for clang

Average Execution time statistics for the benchmark applications which are given in table 1 with gcc are indicated in table 3 and average execution time statistics for the clang are indicated in table 4:

Average Benchmark Results GCC				
llu	7zip-benchmark	Reduction-flt	is	bullet
12.3607s	8.8190s	5.3859s	5.6417s	3.5373s

Table 3: Average execution time statistics for gcc

Average Benchmark Results Clang				
llu	7zip-benchmark	Reduction-flt	is	bullet
9.0986s	9.5644s	5.6190s	4.9309s	5.5134s

Table 4: Average execution time statistics for clang

Visualization of benchmark results and their comparison in gcc and clang are as given in the following figure:

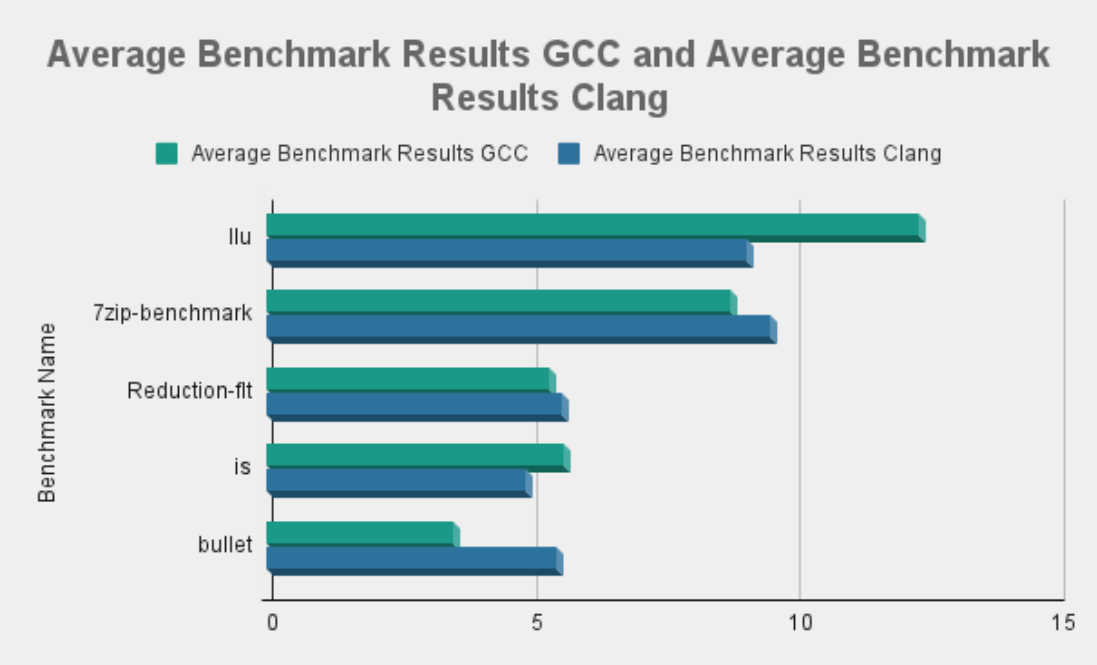


Figure 1: GCC and Clang benchmark execution time comparison

As we can observe from the execution of benchmarks give in the figure 1 for the applications that I have chosen, gcc has shorter execution time on average for three of the benchmarks and clang has shorter execution time on average for two of the benchmarks.

References

[1] URL: https://llvm.org/docs/GettingStarted.html.

[2] Fernando Magno Pereira. *What is LLVM?* Dec. 2020. URL: https://www.youtube.com/watch?v=HecW5byOrUY&list=PLDSTpI7ZVmVnvqtebWnnI8YeB8bJoGOyv&index=1.

[3] Fernando Magno Pereira. *Installing LLVM*. Dec. 2020. URL: https://www.youtube.com/watch?v=l0LI_7KeFtw&list=PLDSTpI7ZVmVnvqtebWnnI8YeB8bJoGOyv&index=2.

[4] URL: https://llvm.org/docs/TestSuiteGuide.html.