Linux OpenMP Benchmark Results

Introduction

The purpose is to test 2 benchmarks to compare performance in terms of MFLOPS (Million Floating Point Operations Per Second) by executing them with and without OpenMP on both 32 bit as well as 64 bit Linux architectures. The 2 benchmarks used are MemSpeed and Original OpenMP.

MemSpeed-

It consists of three different sequences of operations on 64 bit double precision floating point numbers, 32 bit single precision floating point numbers and 32 bit integers via 2 data arrays. The types of operations are-
x[i] = x[i] + c * y[i]
x[i] = x[i] + y[i]
x[i] = y[i]

The MFLOPS performance measure is calculated by dividing the MB/s for double and single precision floating point numbers by 8 and 4 respectively. On the other hand, the same measure is obtained for integers by multiplying the MB/s with 11 / 8 since both 64 and 32 bit versions are translated to 11 integer instructions per 8 data words.

Original OpenMP-

It consists of arithmetic operations of the form x[i] = (x[i] + a) * b – (x[i] + c) * d + (x[i] + e) * f with 2, 8 or 32 operations per input data word. Array sizes used are 0.1, 1 or 10 million 4 byte single precision floating point words.

2 operations-
x[i] = (x[i] + a) * b
8 operations-
x[i] = (x[i] + a) * b – (x[i] + c) * d + (x[i] + e) * f
32 operations-
x[i] = (x[i] + a) * b – (x[i] + c) * d + (x[i] + e) * f – (x[i] + g) * h + (x[i] + j) * k – (x[i] + l) * m + (x[i] + o) * p – (x[i] + q) * r + (x[i] + s) * t – (x[i] + u) * v + (x[i] + w) * y

Without OpenMP, both the benchmarks run on a single CPU, with OpenMP, both the benchmarks' workloads get distributed over all the cores of all configured CPUs.

Setting up the environment

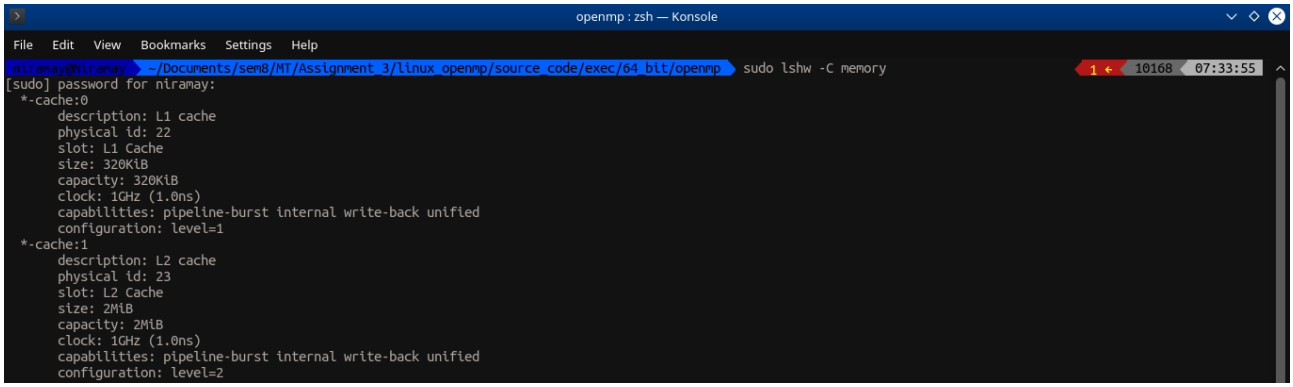
Requirements-

Linux system
OpenMP support
GCC compiler
Python3 with matplotlib module (for plotting the performance comparison graphs)
System monitor (for observing CPU usage)

System specifications-

Linux 4.15.0-88-generic
16.04.1-Ubuntu x86_64
AMD A10-9600P RADEON R5, 10 COMPUTE CORES 4C+6G

CPUs 4, Configured CPUs 4
RAM Size 11.13 GB, Page Size 4096 Bytes

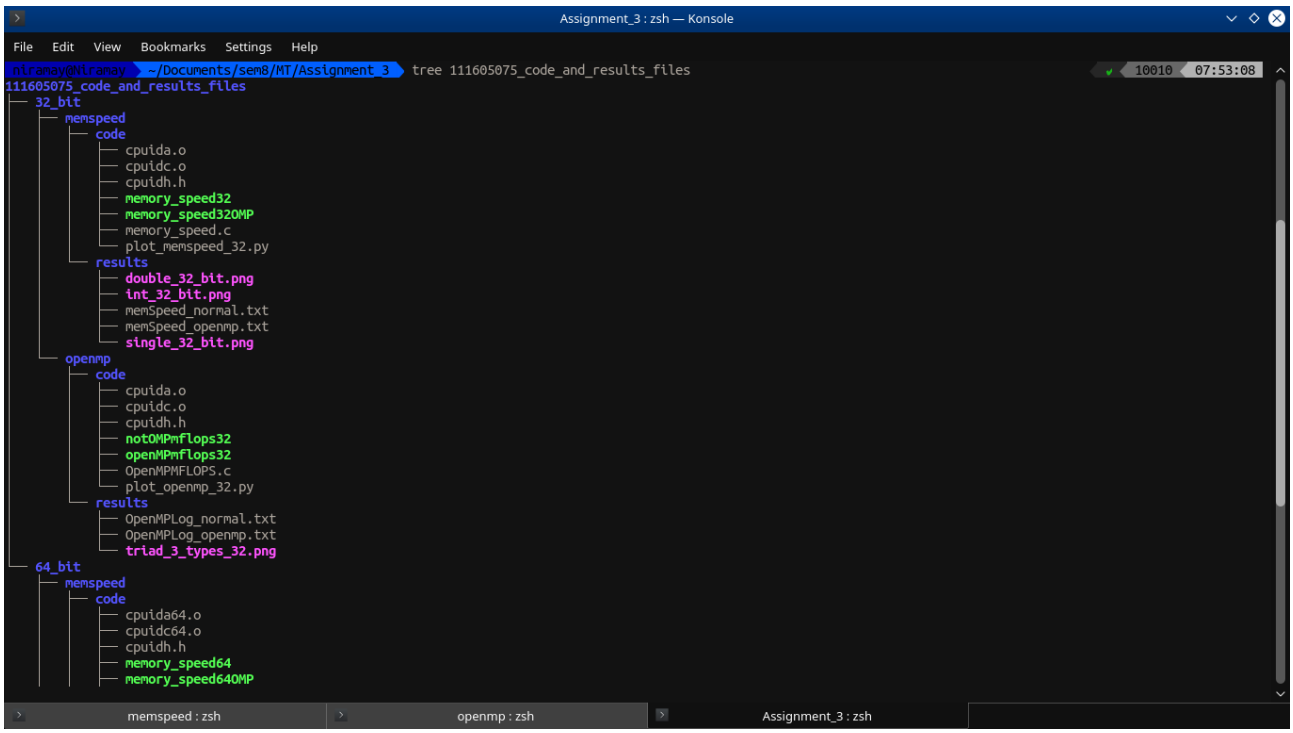


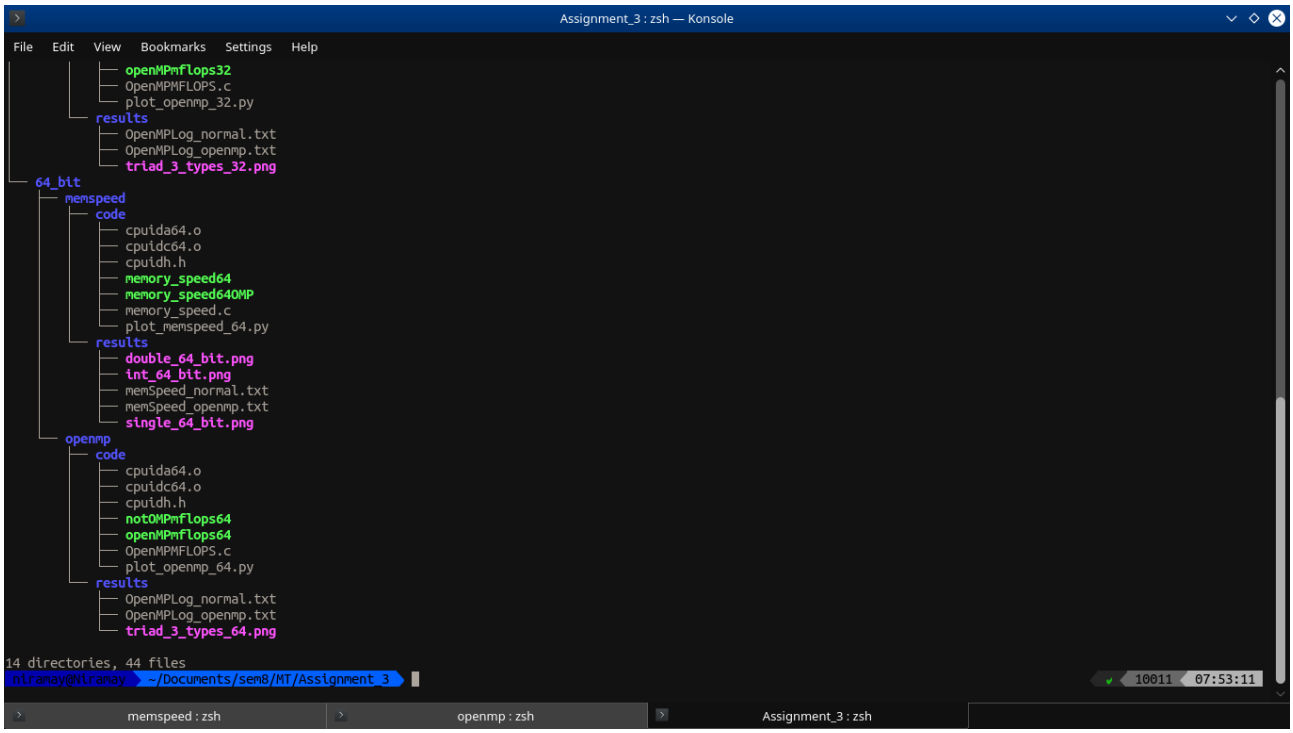
Problem statement

Measuring performance of Linux OpenMP benchmarks in terms of MFLOPS upon execution with and without use of OpenMP, and comparing performance in both these scenarios by plotting comparison graphs.

Program

Refer the following tree to locate program files in the submission.

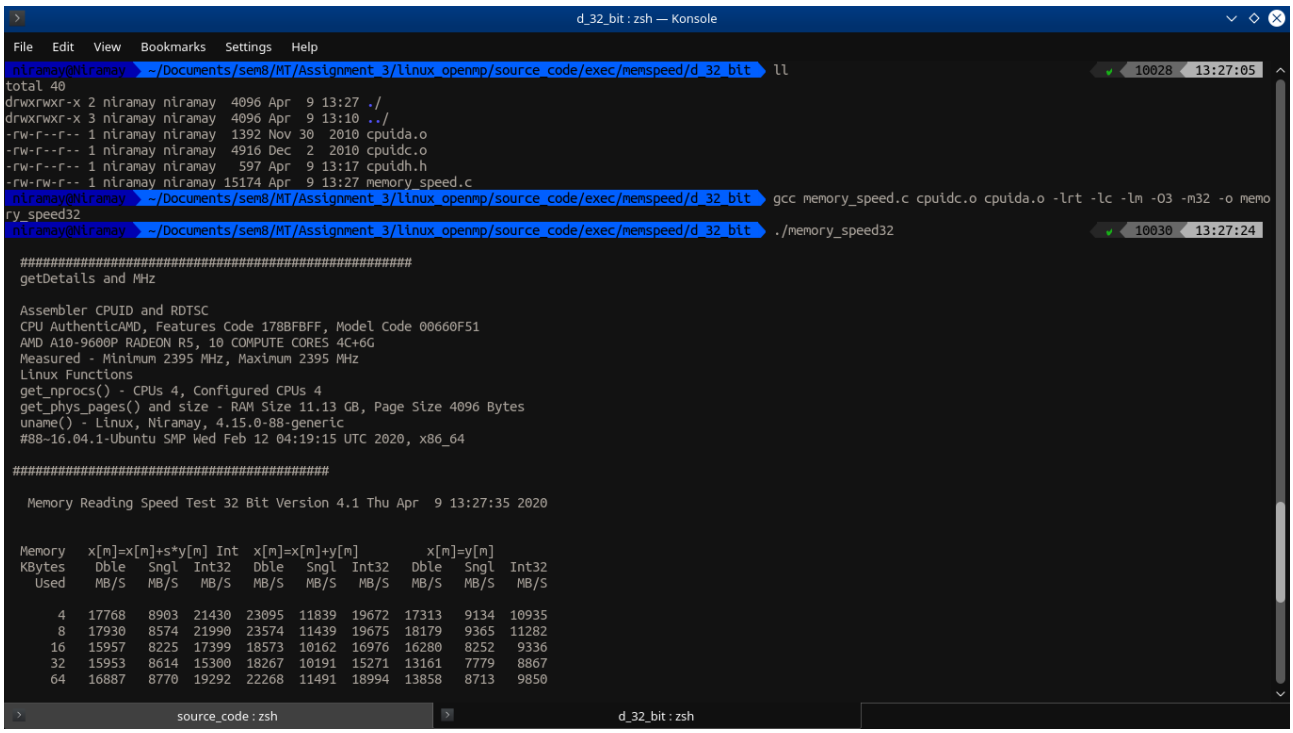


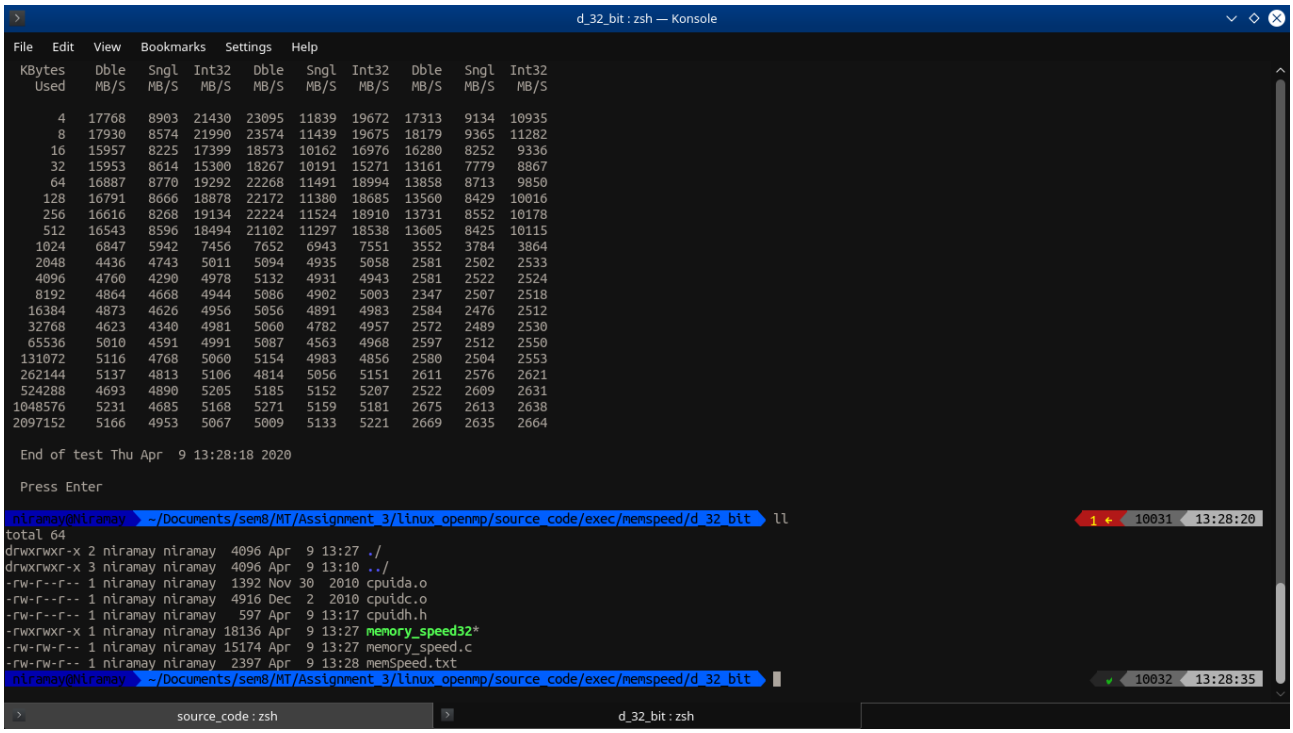


Results

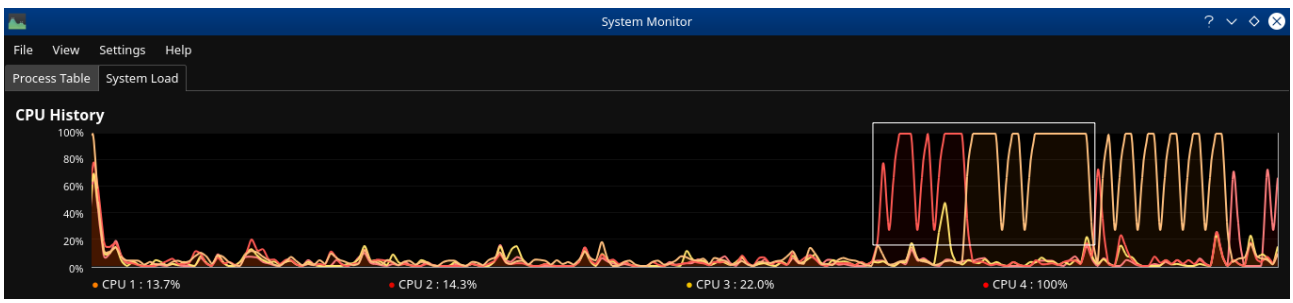
32 bit MemSpeed-

Normal (without OpenMP)

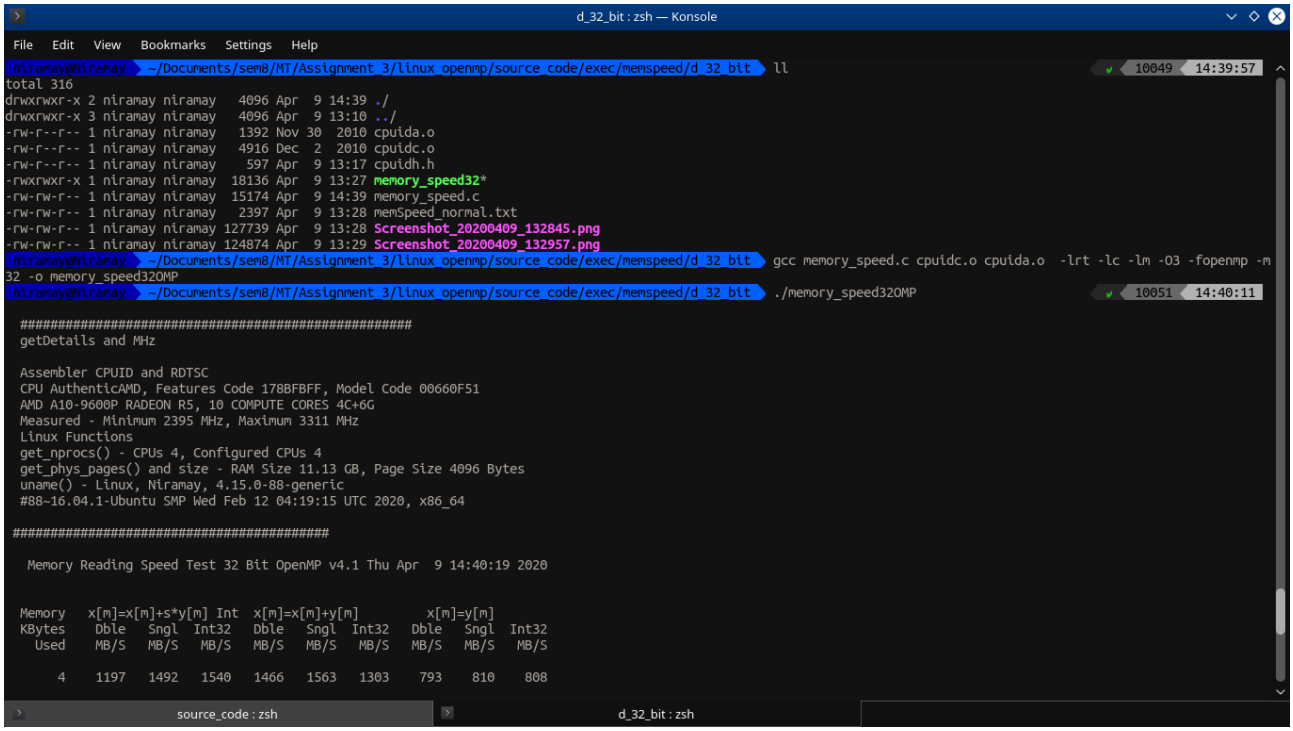


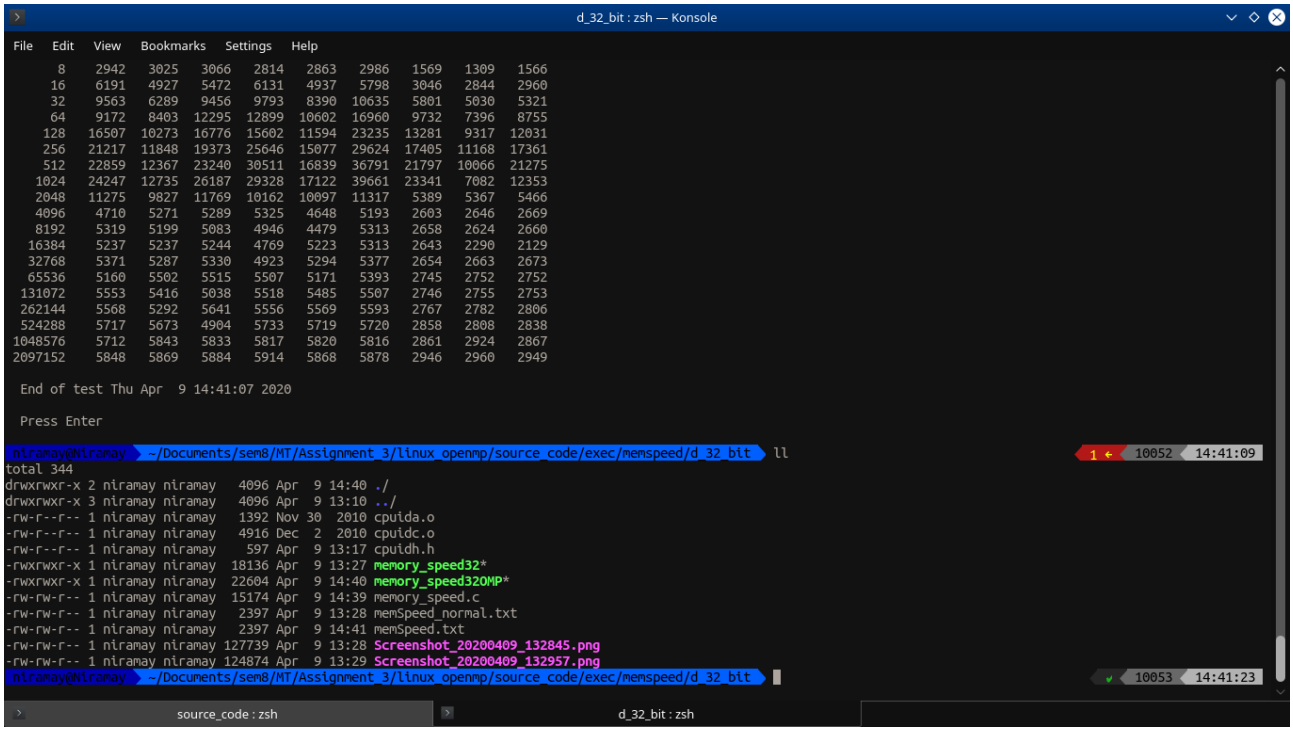


System monitor (single CPU usage, highlighted using a white box)

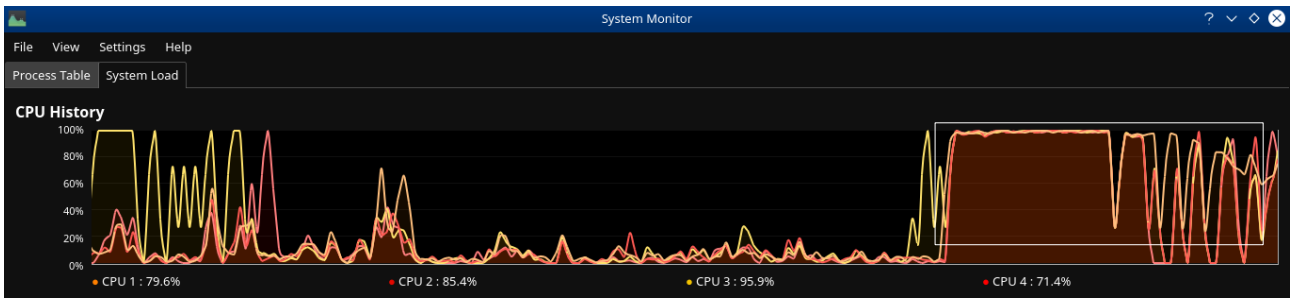


With OpenMP

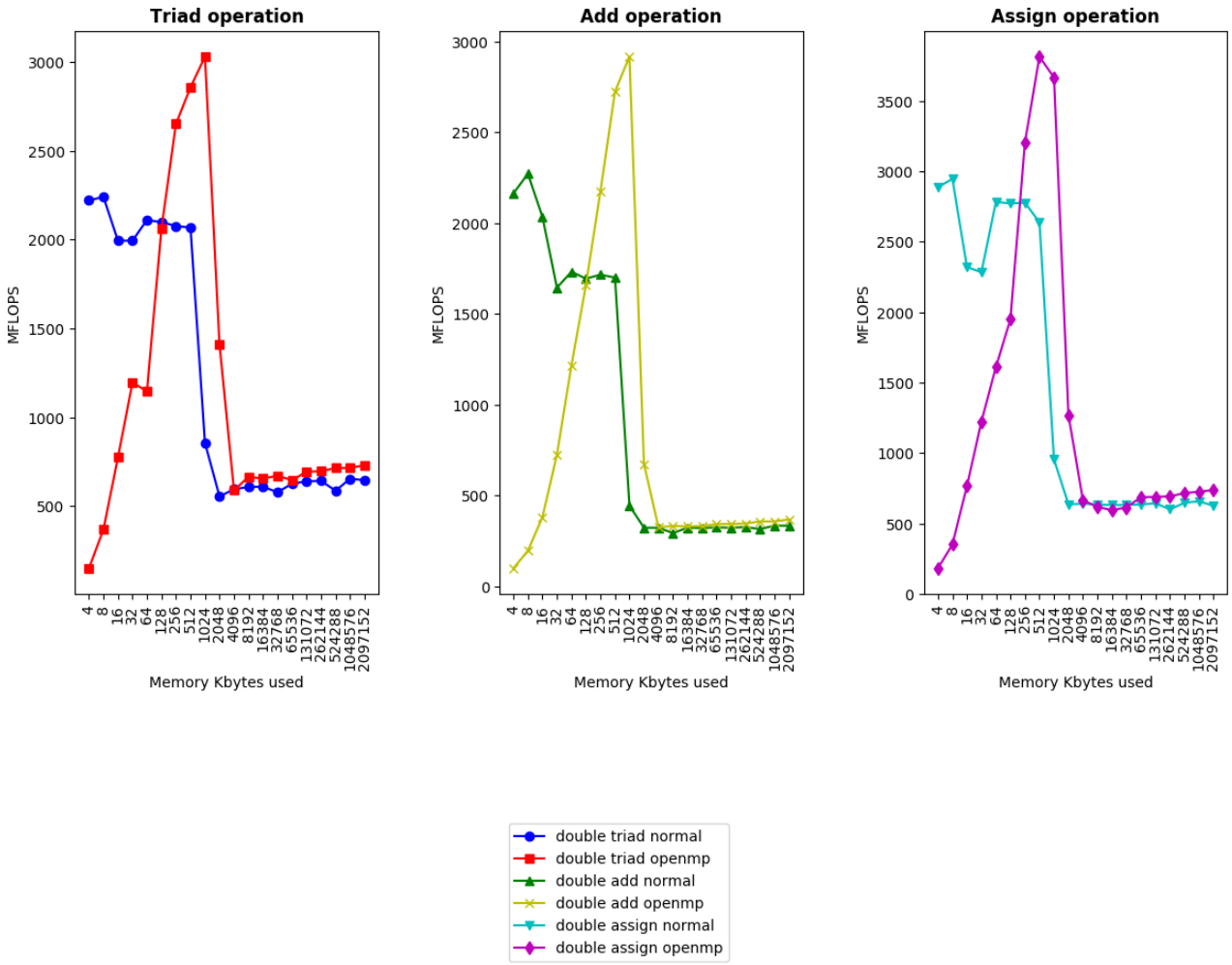


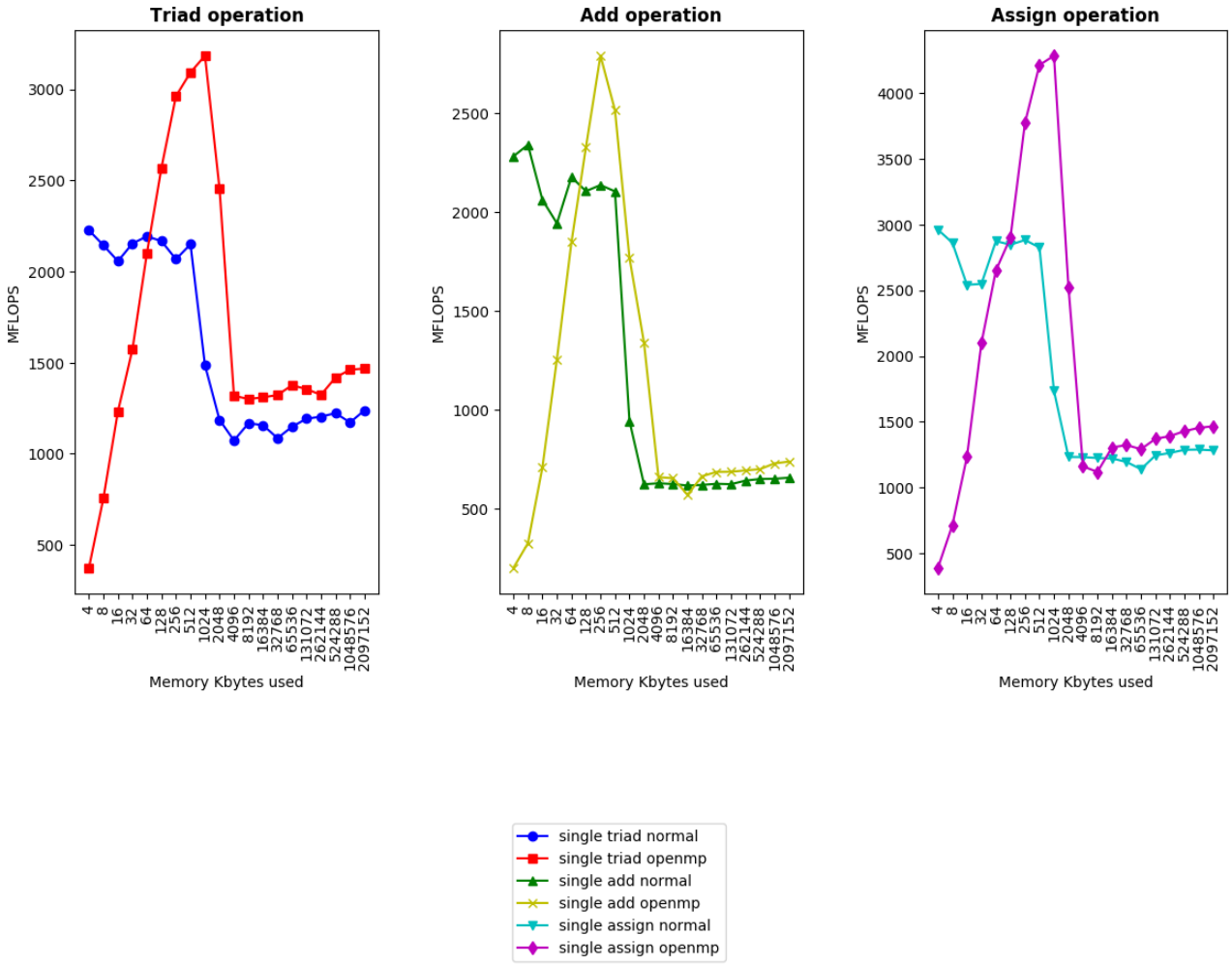


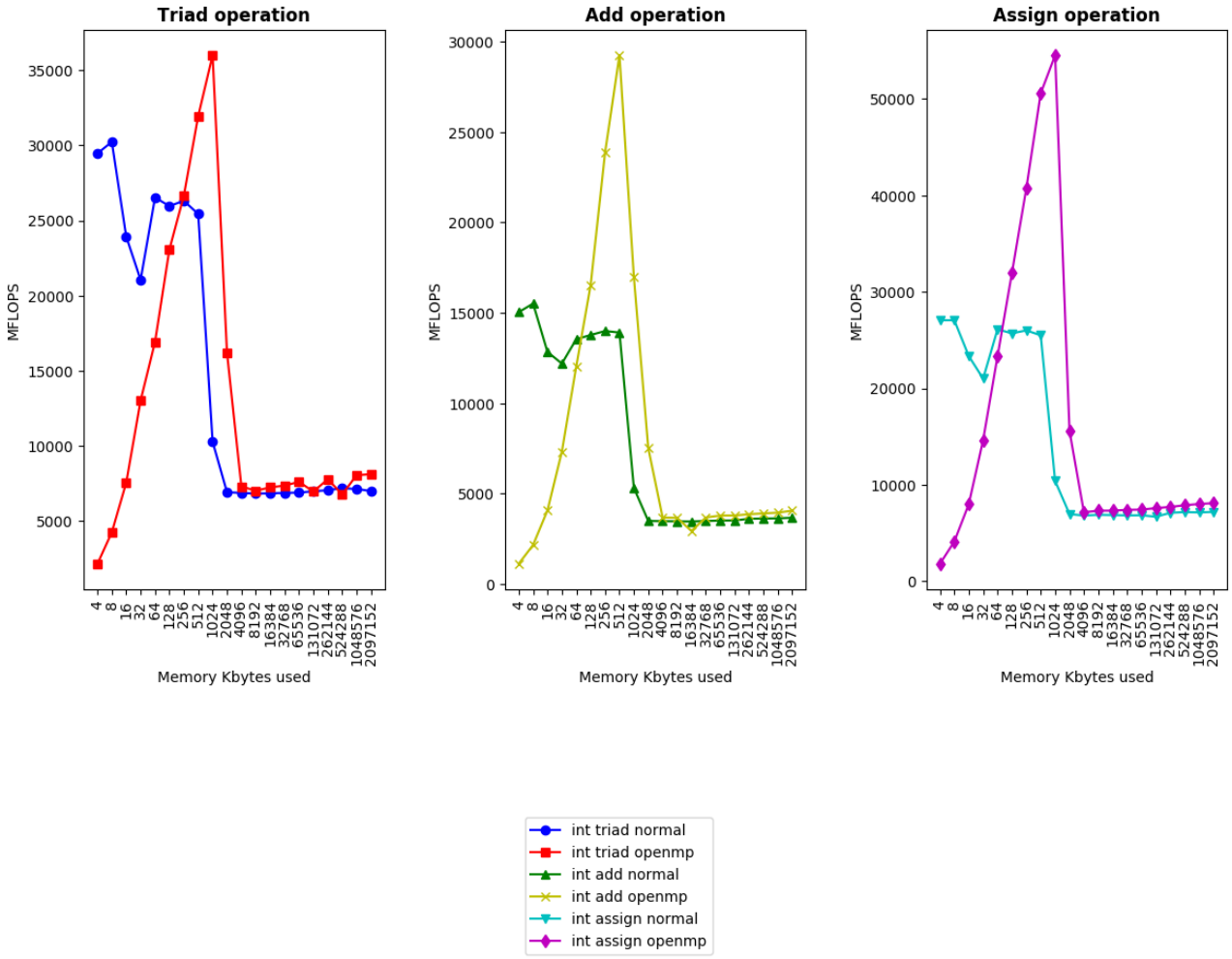
System monitor (usage of all i.e. 4 CPUs, highlighted using a white box)



Performance comparison in MFLOPS

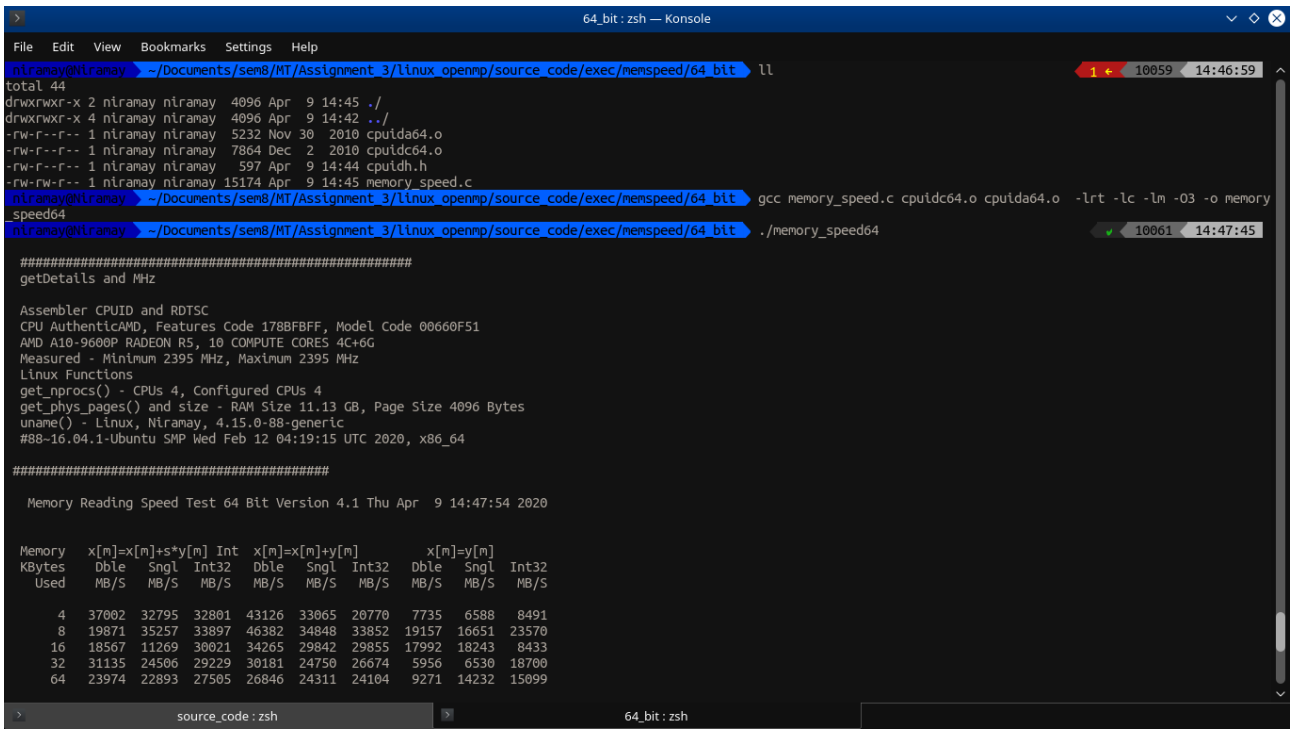


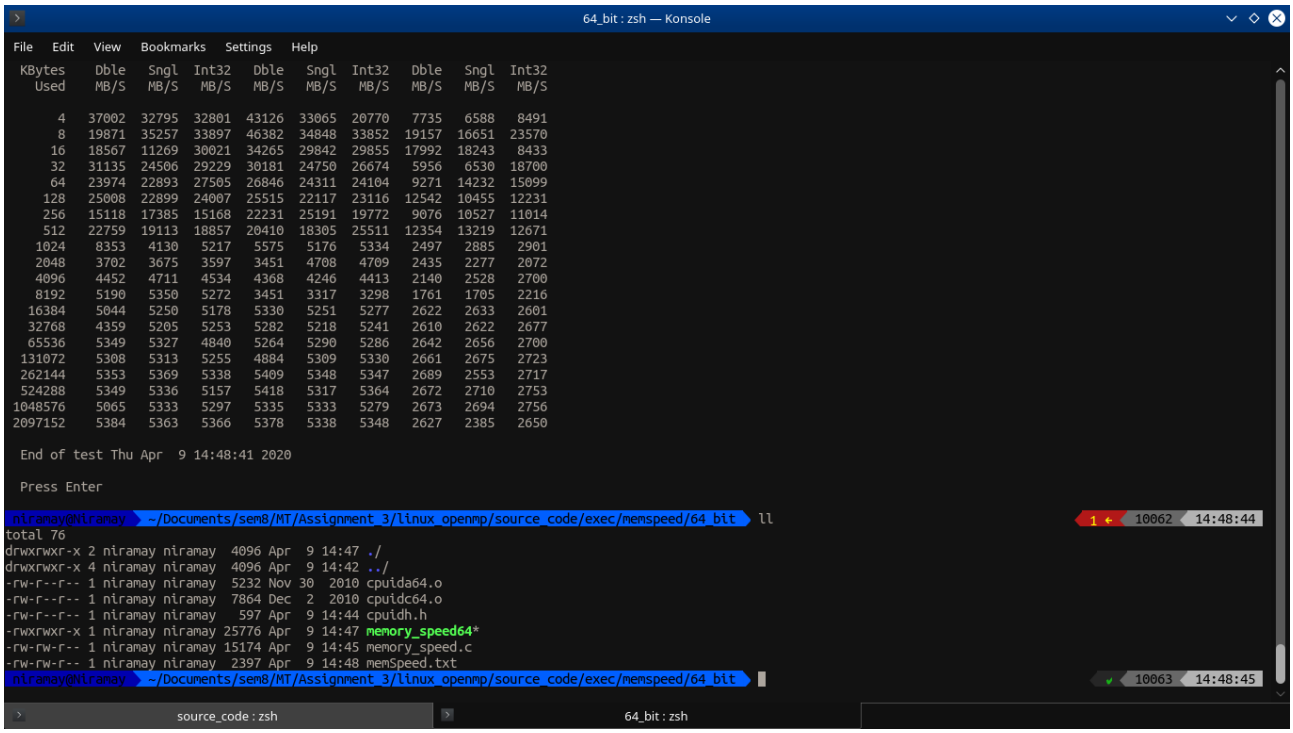

Triad operation
Add operation
Assign operation
MFLOPS
Memory Kbytes used
4
8
16
32
64
128
256
512
1024
2048
4096
8192
16384
32768
65536
131072
262144
524288
1048576
2097152
single triad normal
single triad openmp
single add normal
single add openmp
single assign normal
single assign openmp



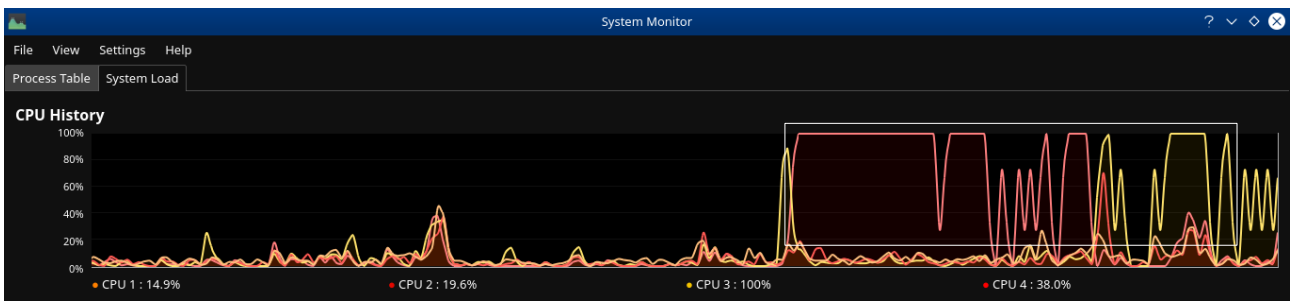
64 bit MemSpeed-

Normal

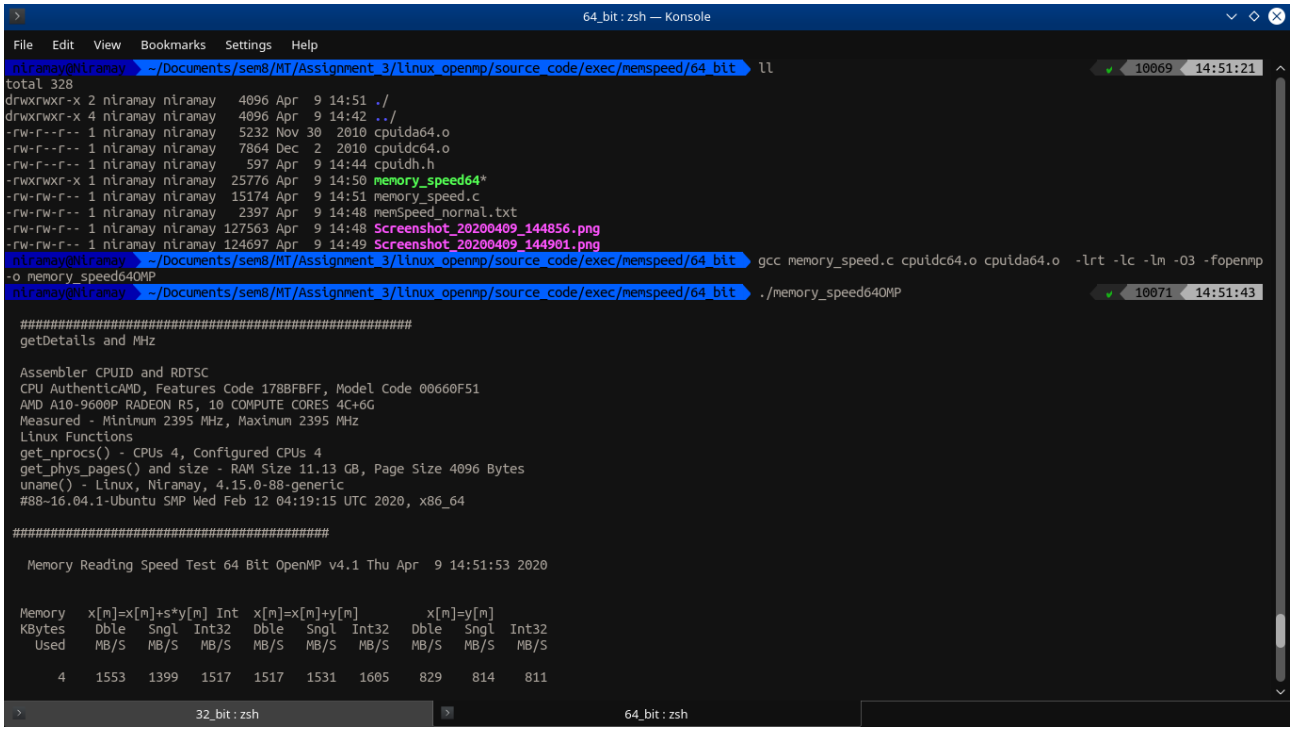


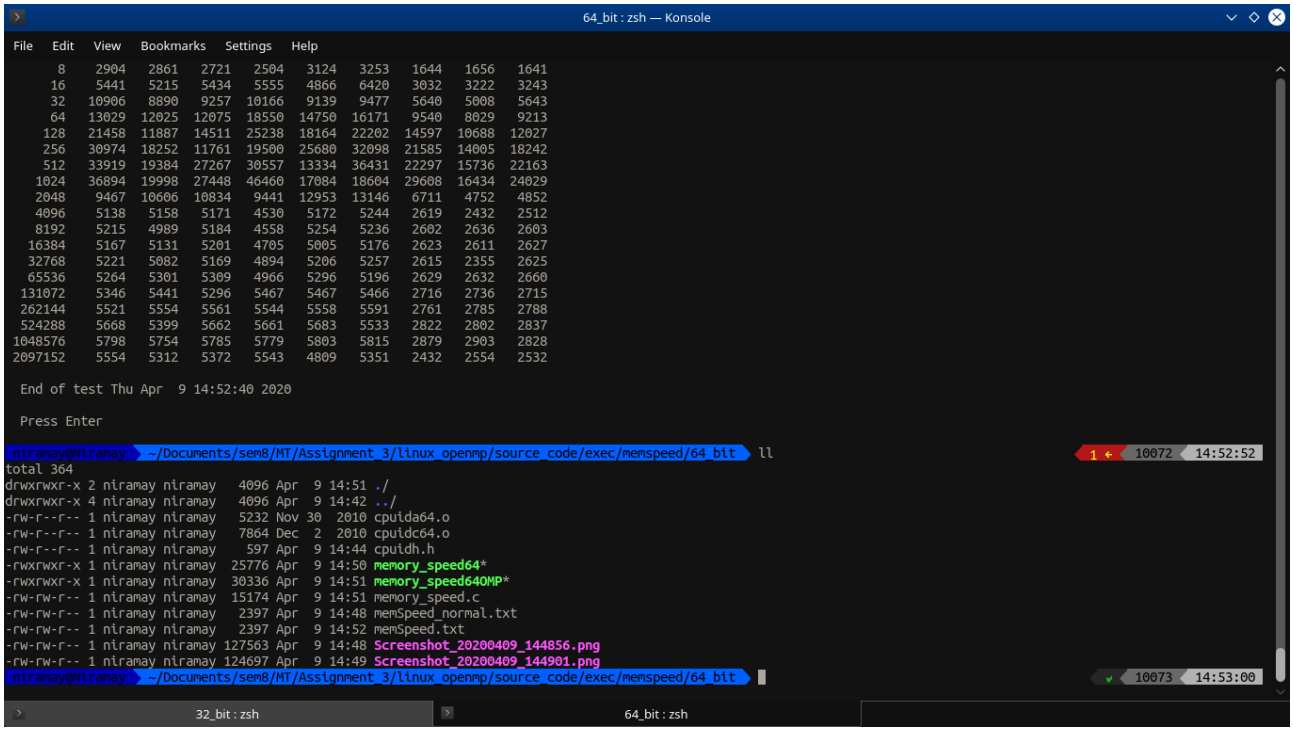


System monitor

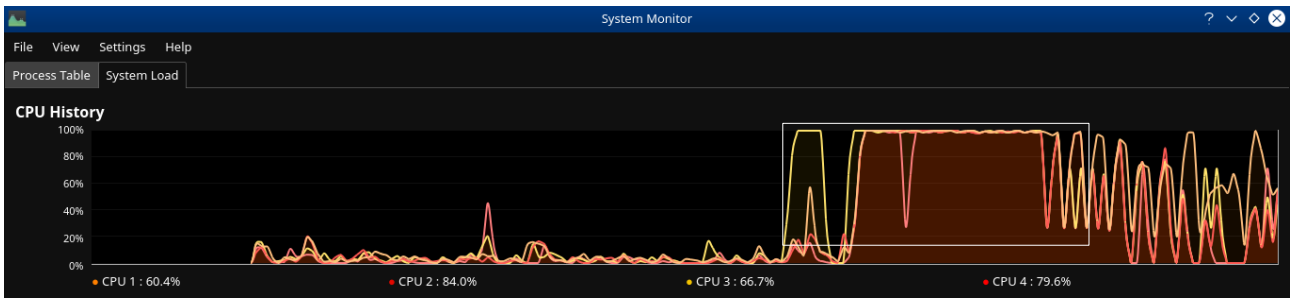


With OpenMP

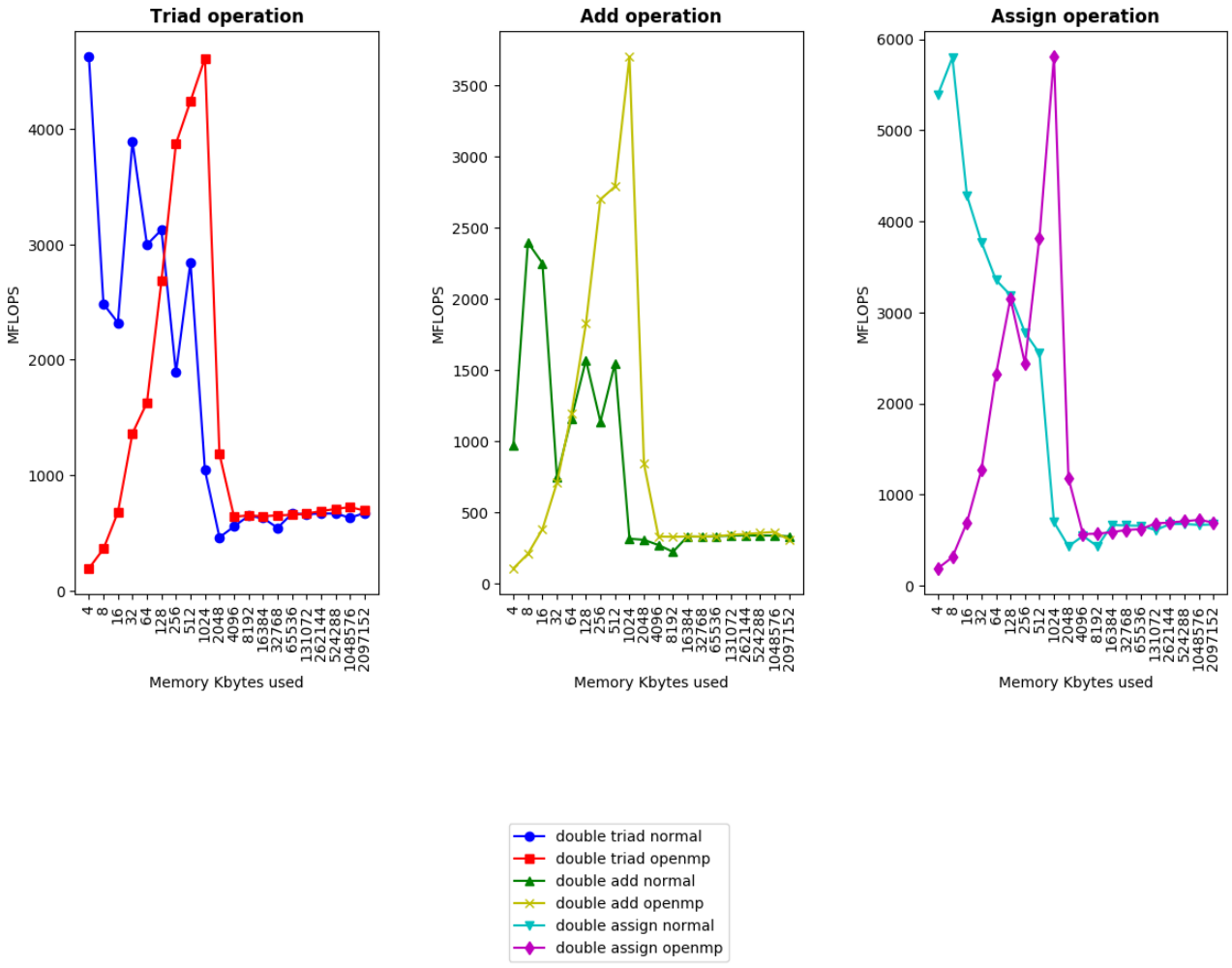


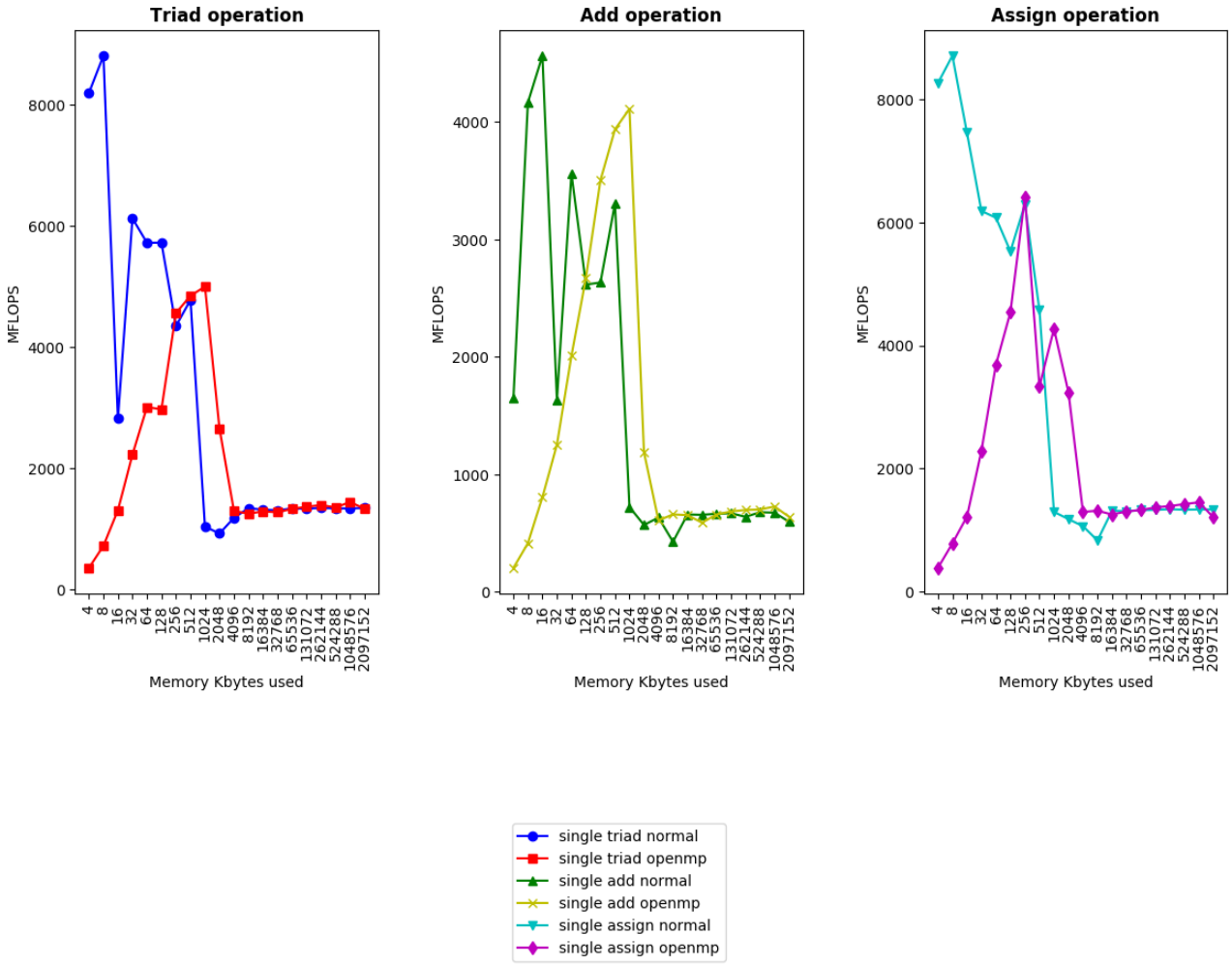


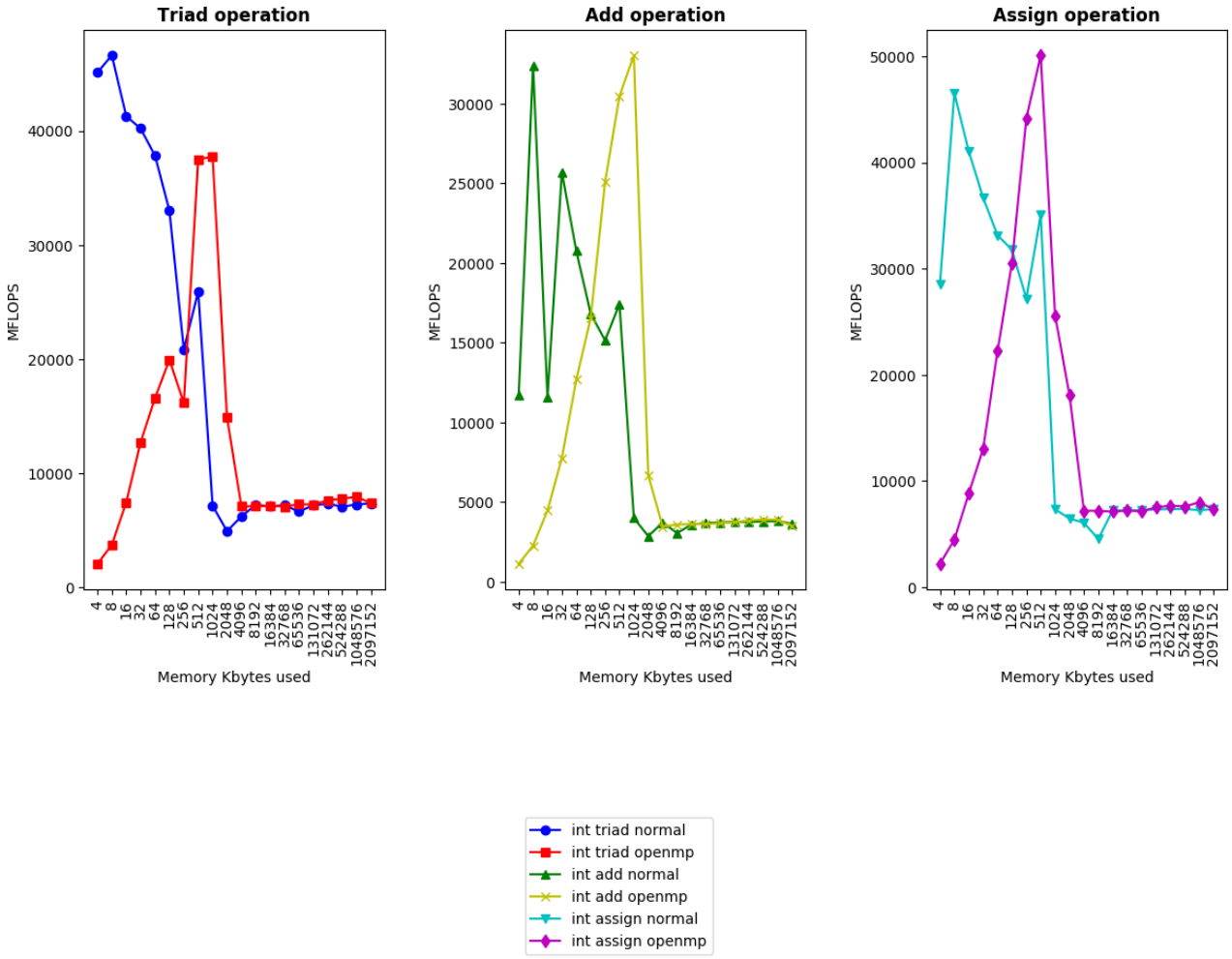
System monitor



Performance comparison in MFLOPS

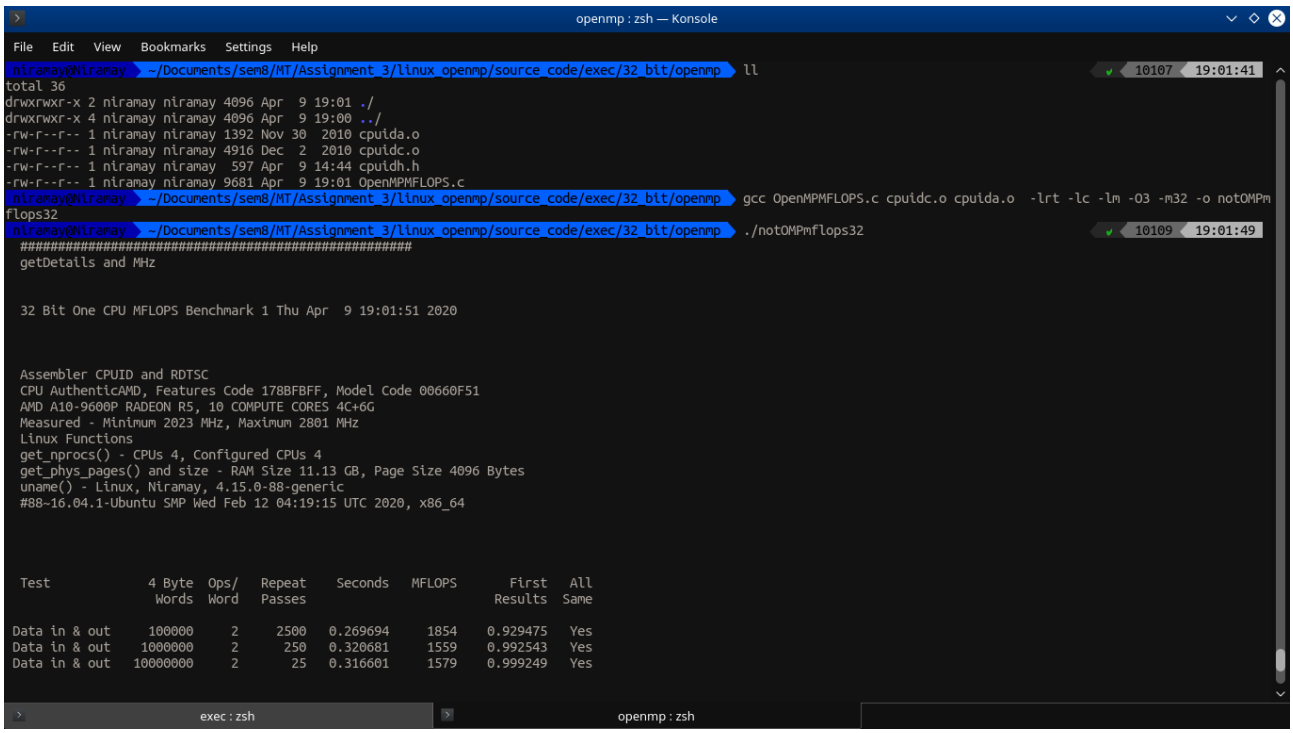


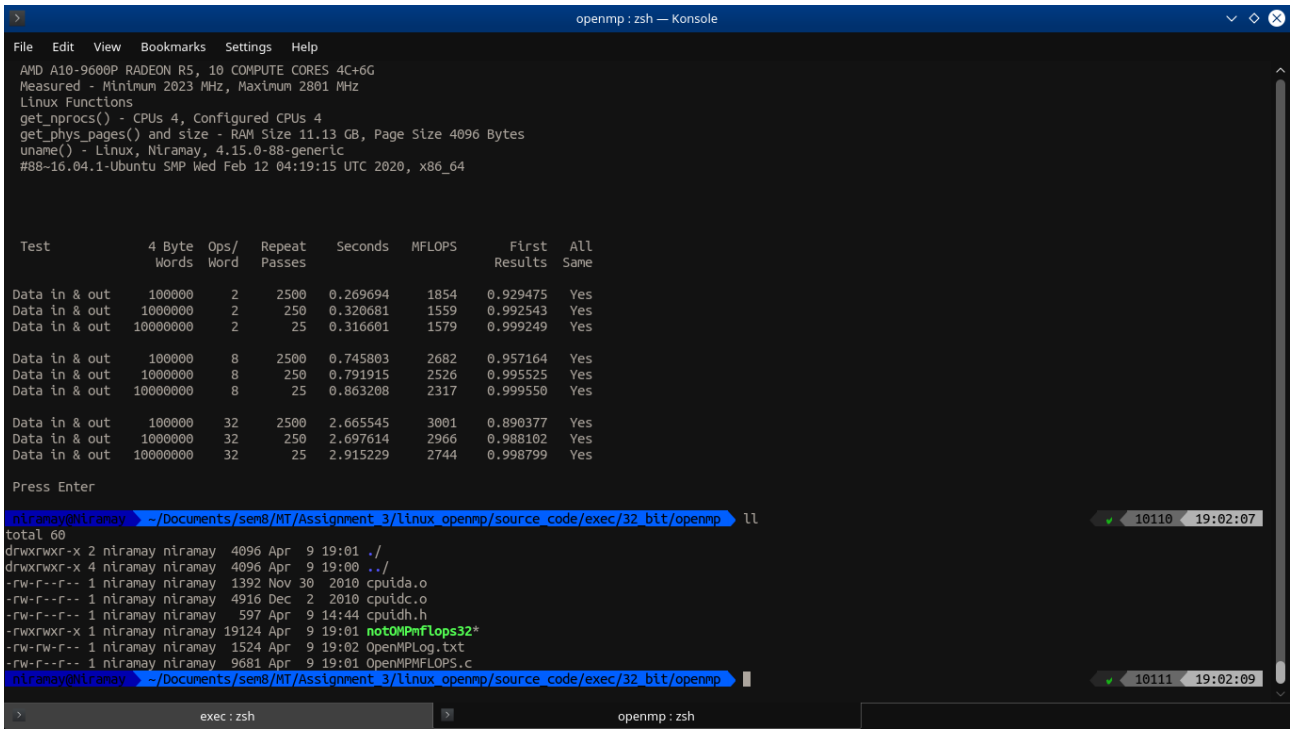

Triad operation
Add operation
Assign operation
MFLOPS
Memory Kbytes used
single triad normal
single triad openmp
single add normal
single add openmp
single assign normal
single assign openmp



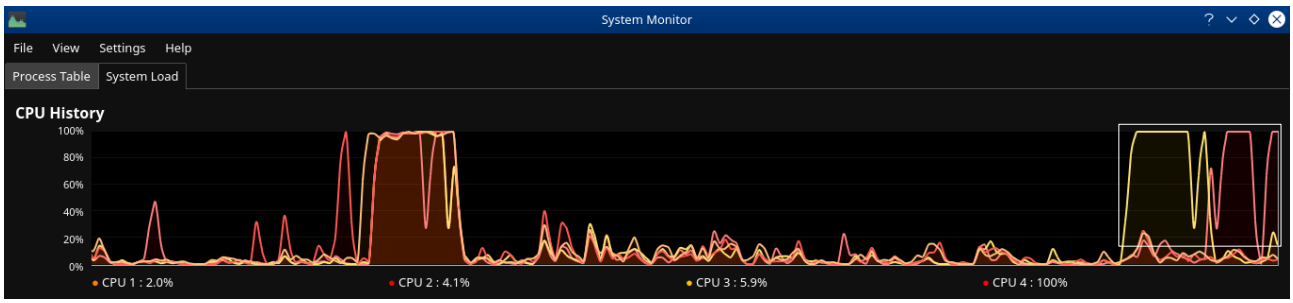
32 bit Original OpenMP-

Normal

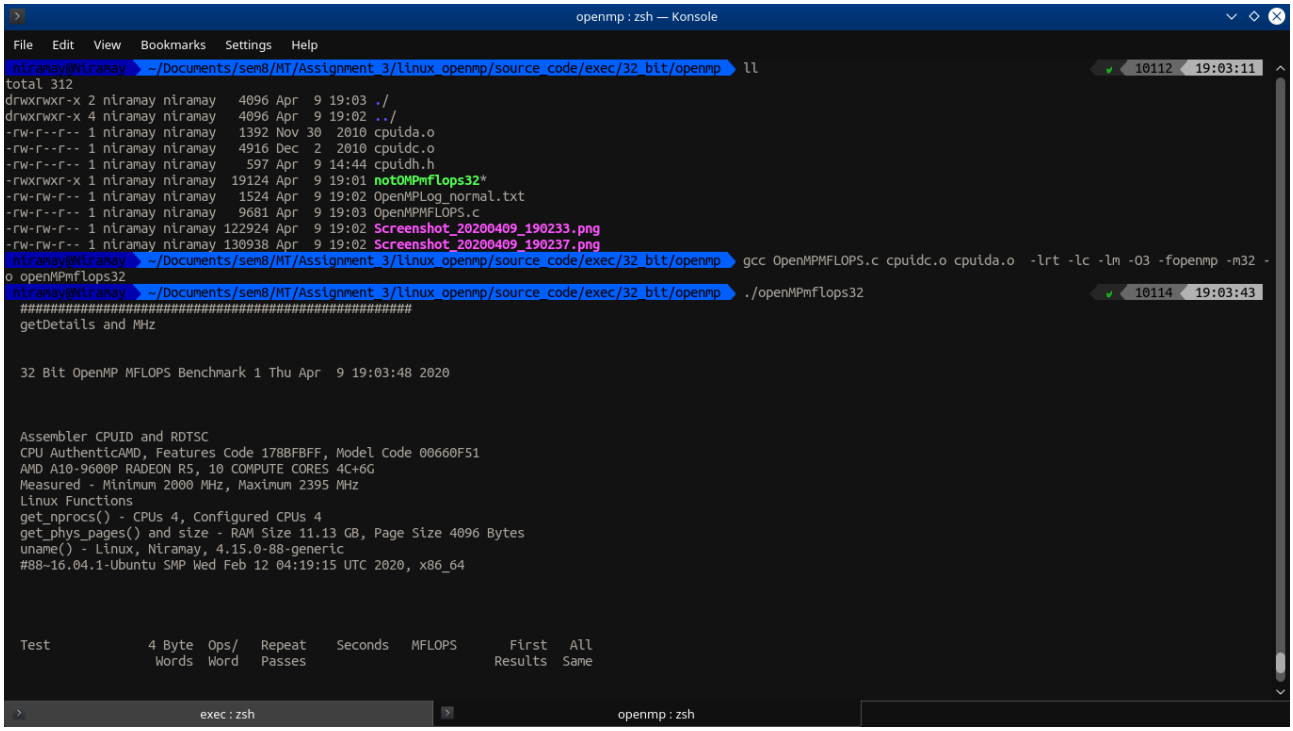


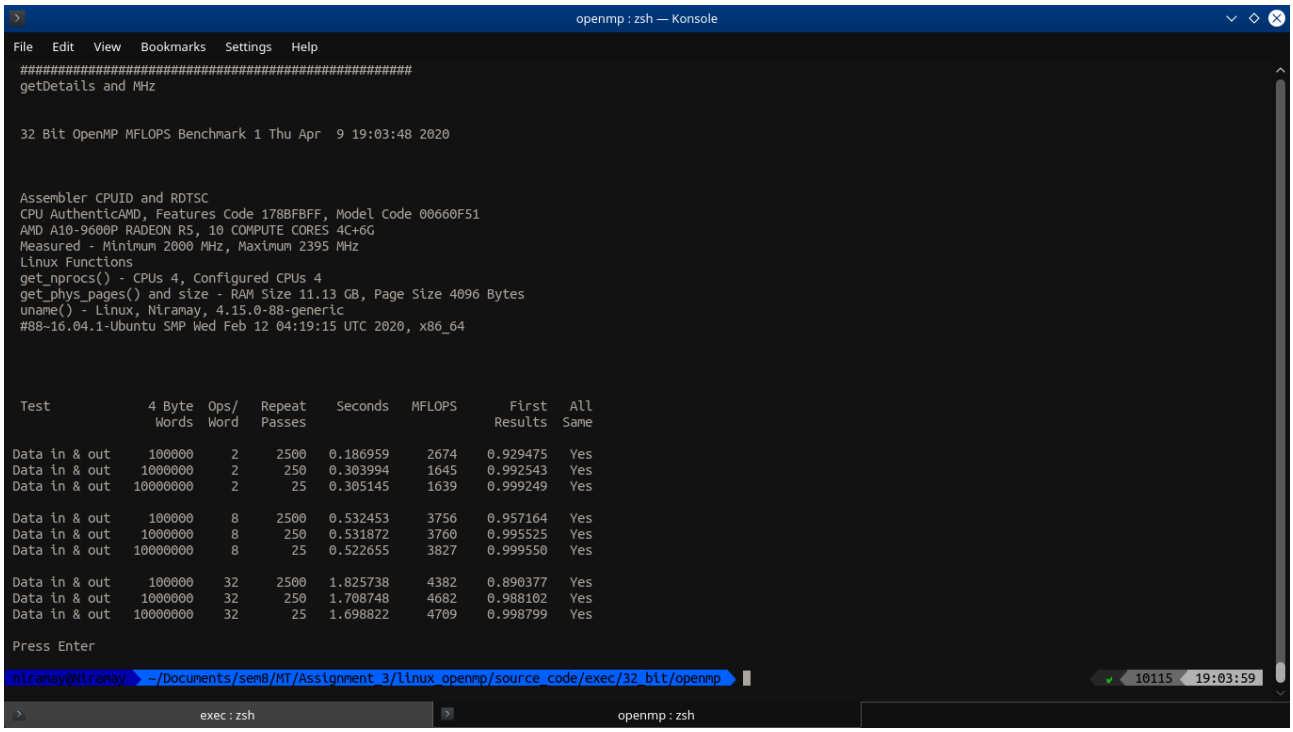


System monitor

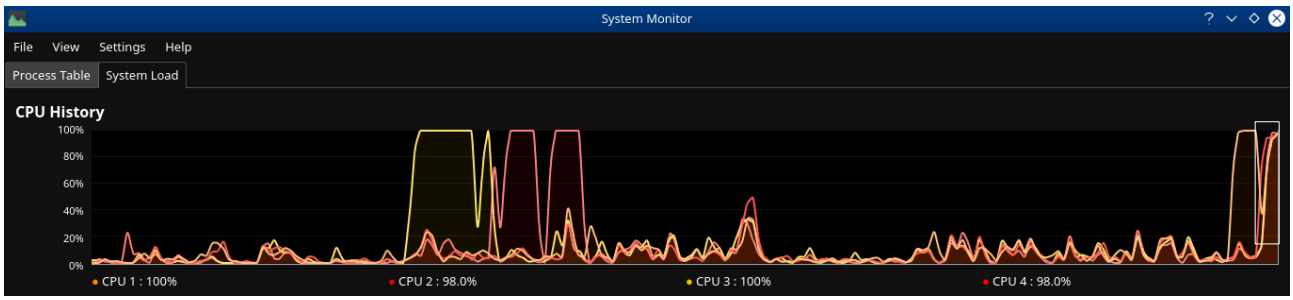


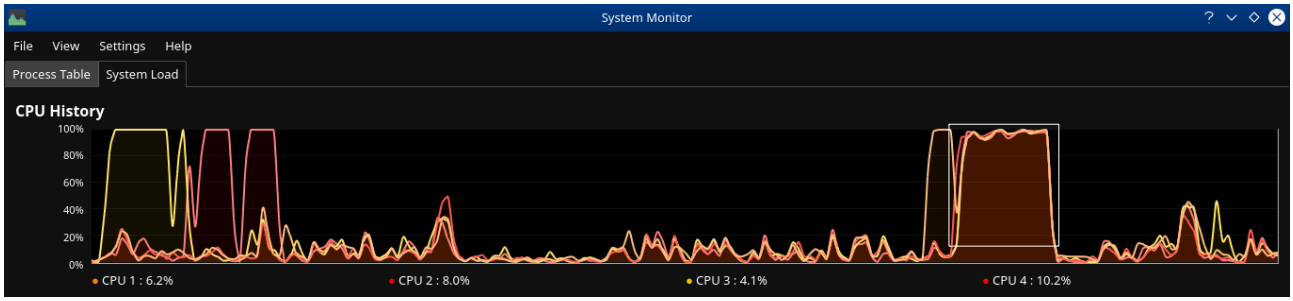
With OpenMP



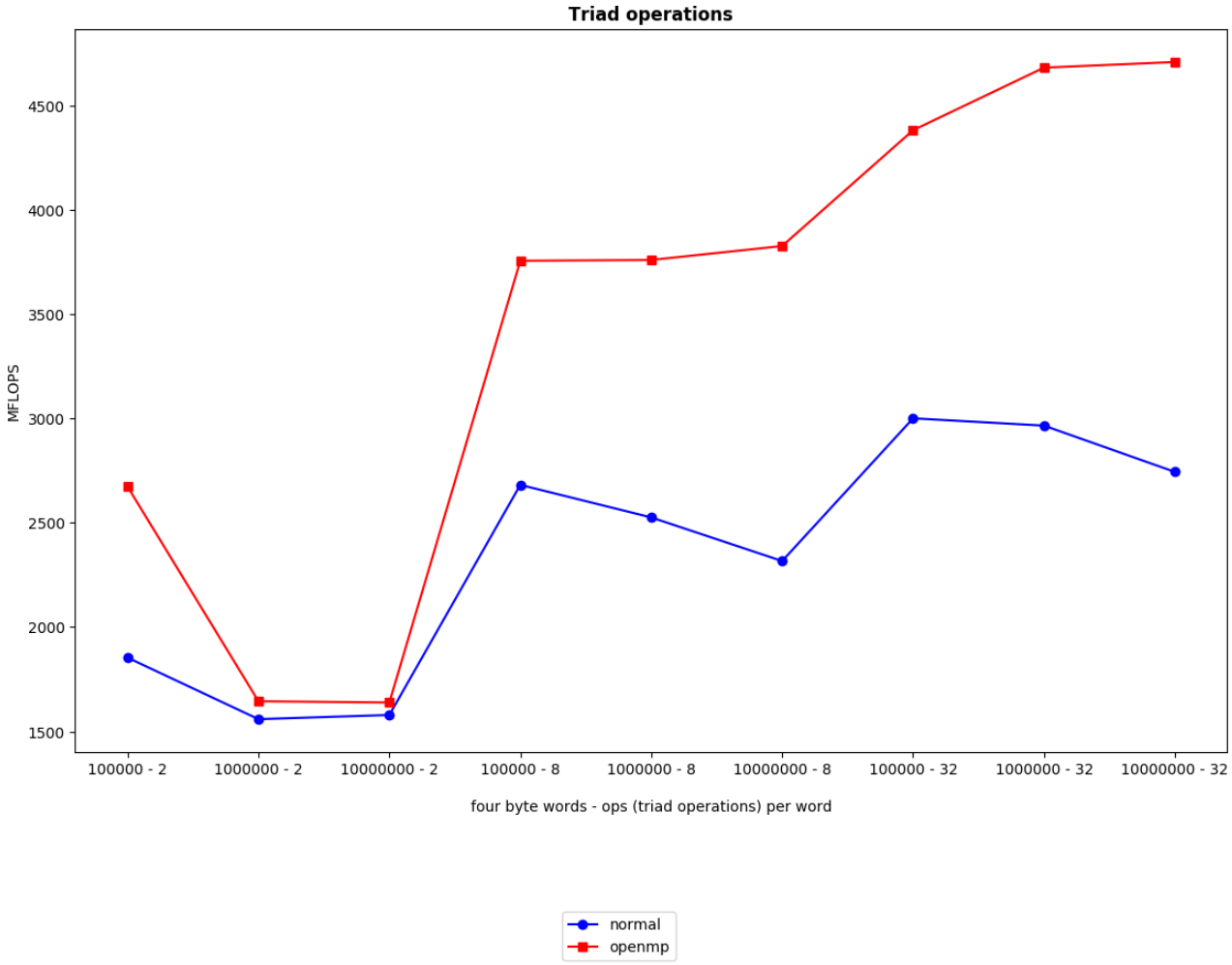


System monitor



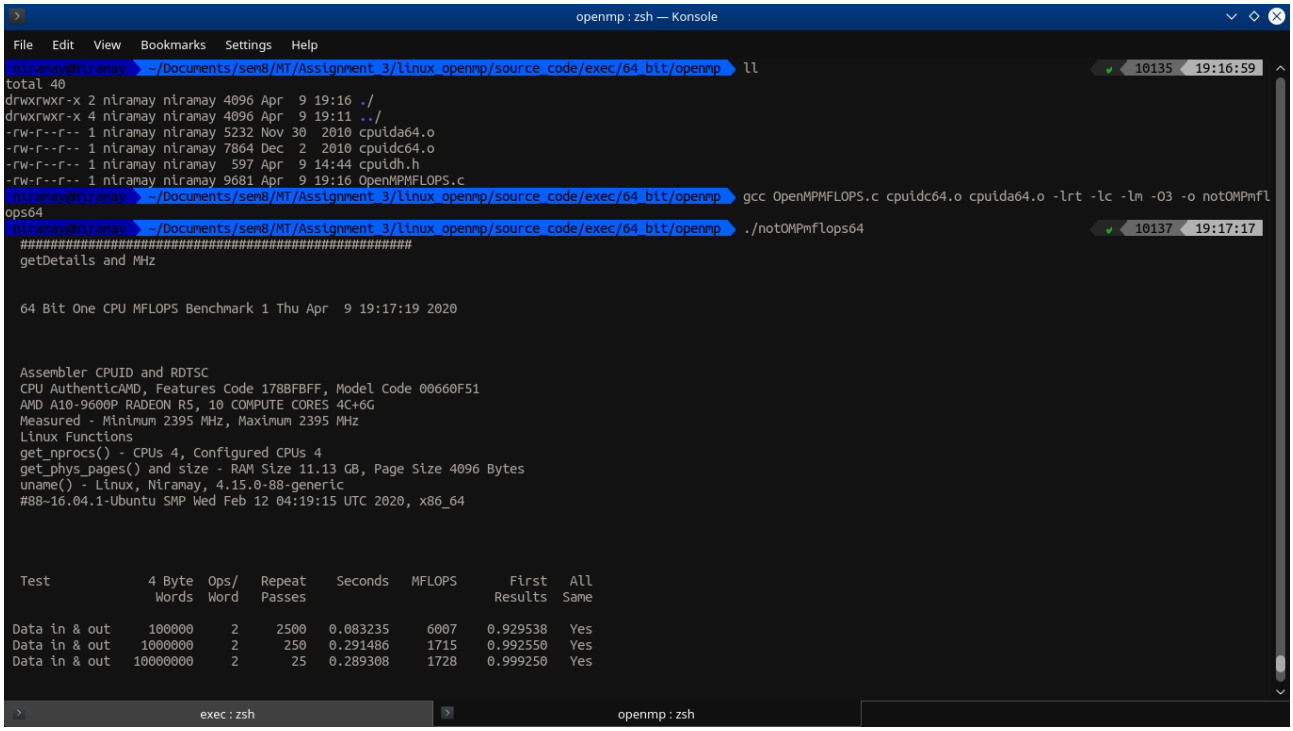


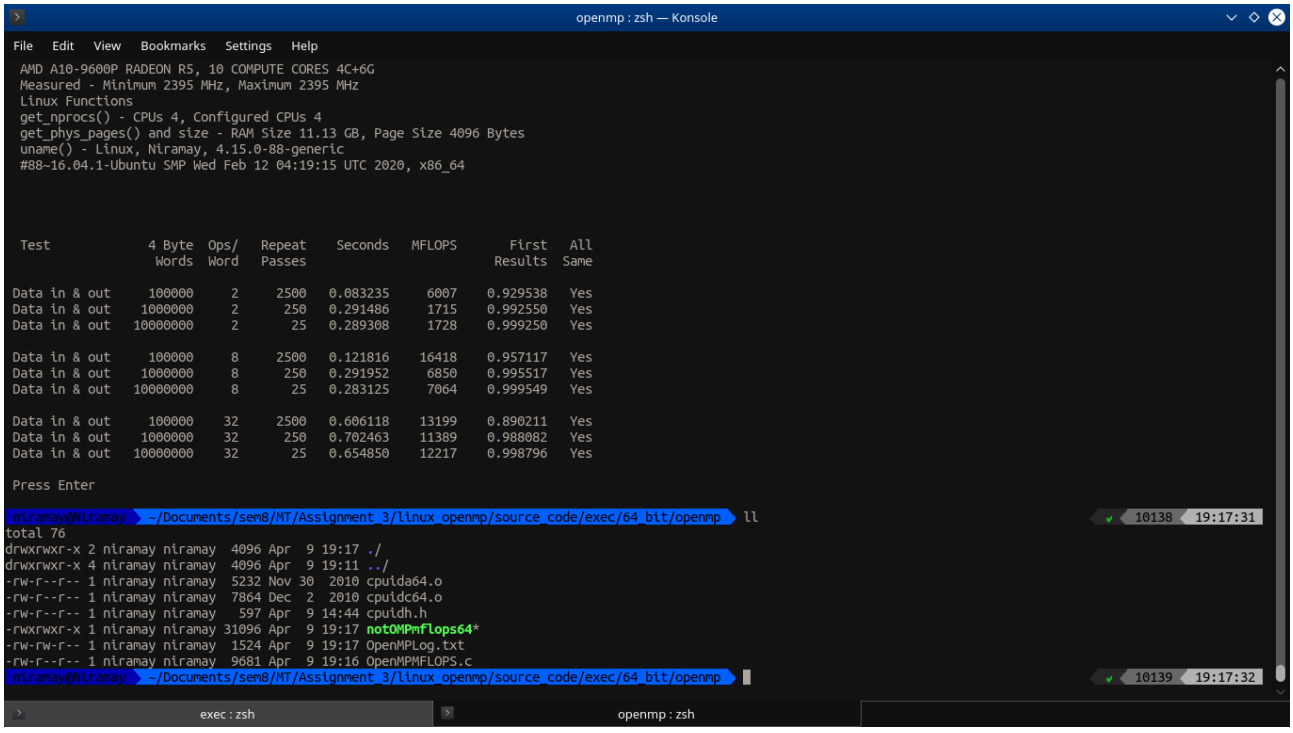
Performance comparison in MFLOPS



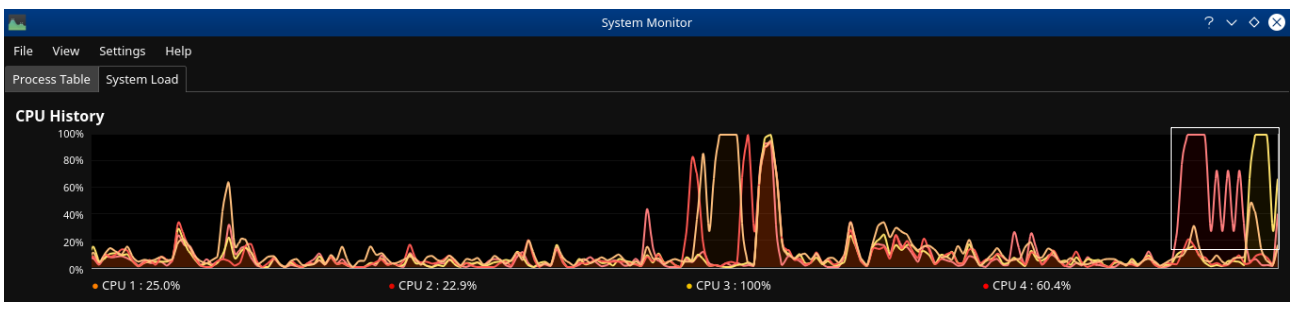
64 bit Original OpenMP-

Normal

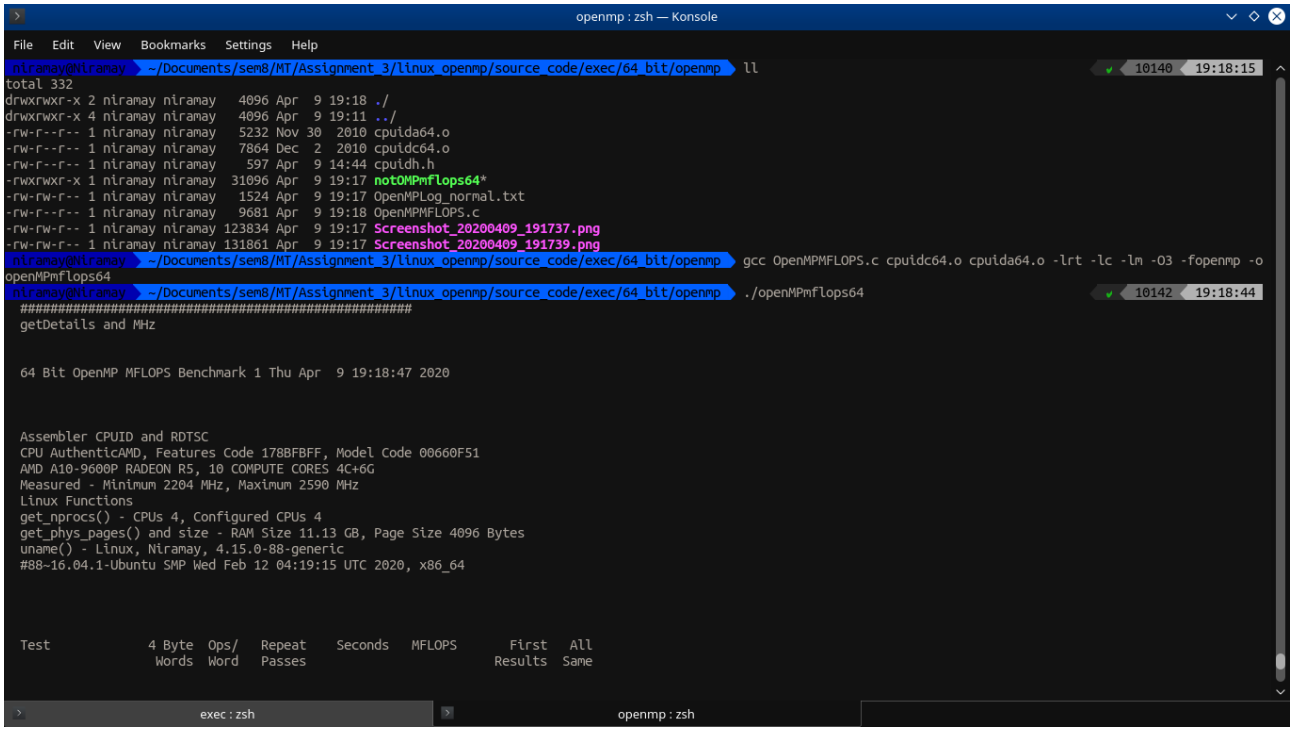


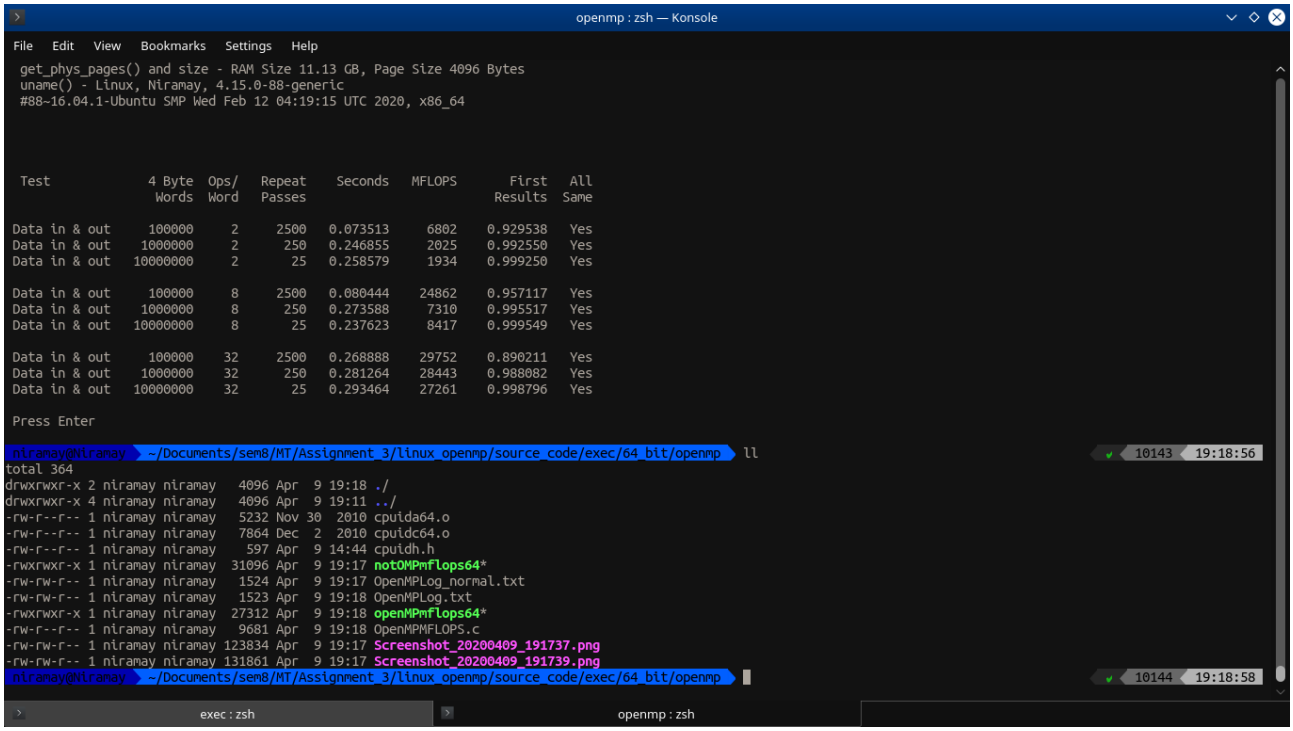


System monitor

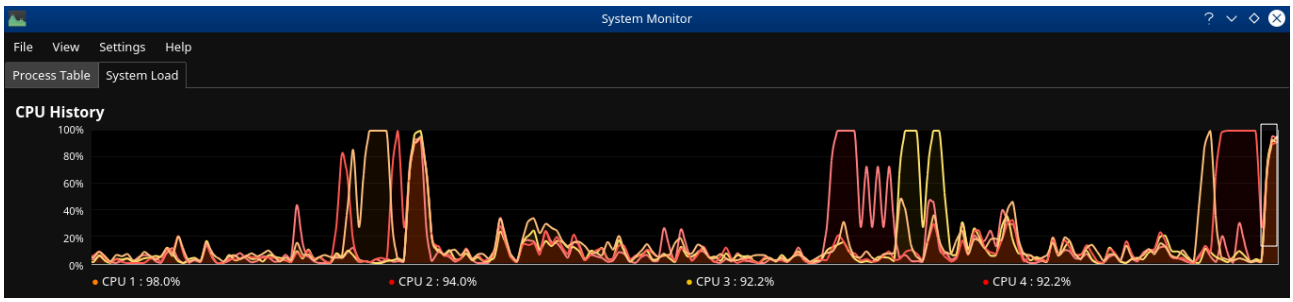


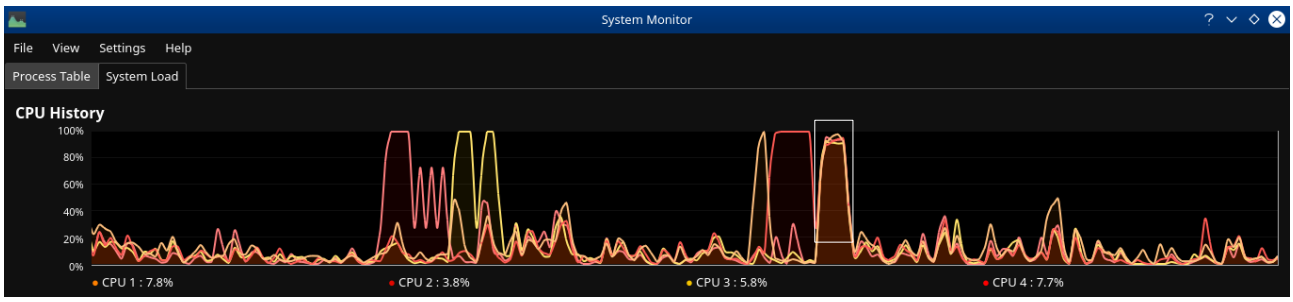
With OpenMP



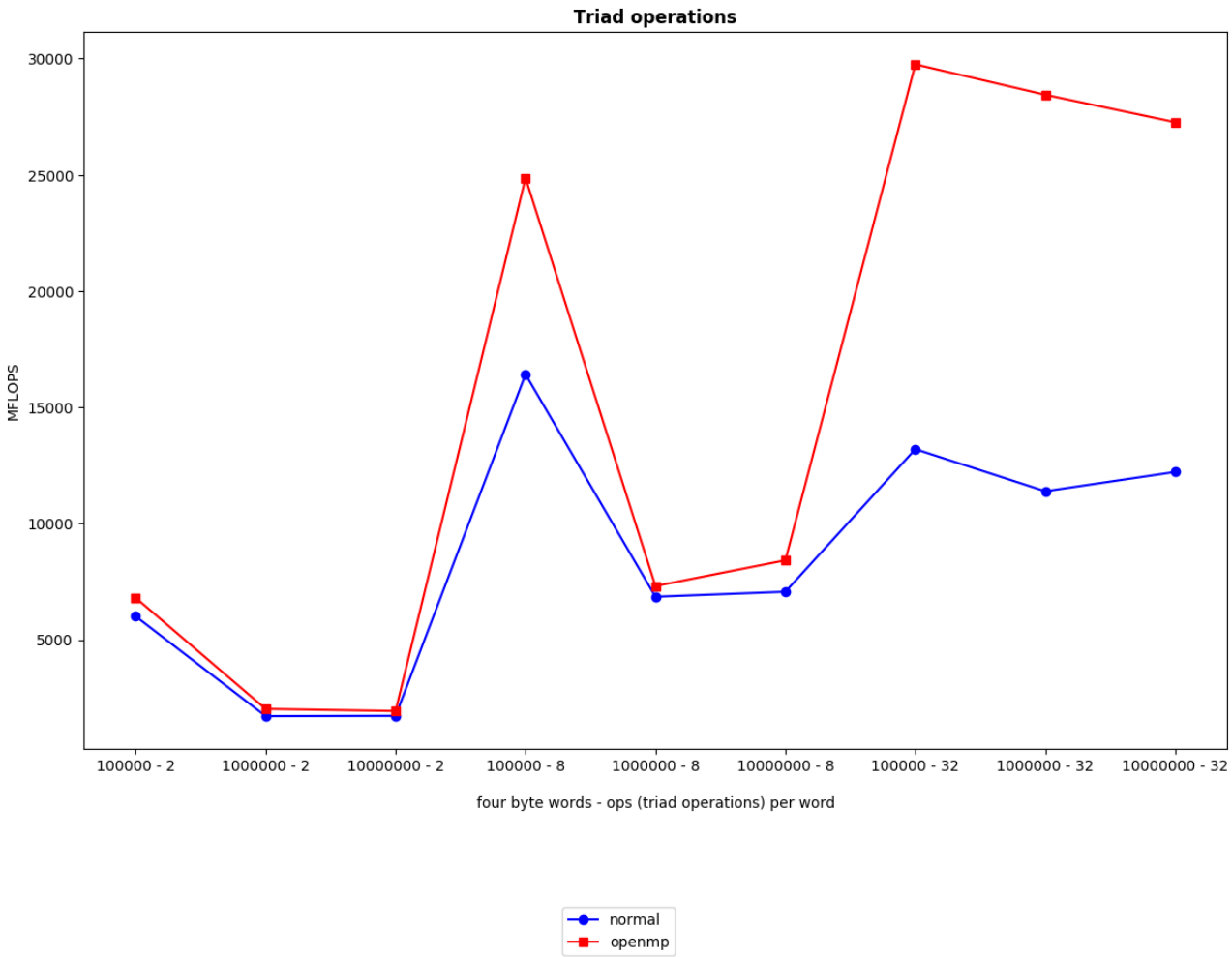


System monitor





Performance comparison in MFLOPS



Conclusion

MemSpeed-

The general observation in case of MemSpeed for both 32 and 64 bit is that execution of the benchmark in terms of MFLOPS performance without OpenMP fares well for low memory usage and then drastically decreases roughly at central memory usage, and then finally remains approximately within a low range with further increase in memory usage. On the other hand, with OpenMP, for low memory usage the performance is really low as compared to without OpenMP, but then it peaks for memory usage roughly in the centre of the low and high limits, and then again drops to match the range without OpenMP with further increase in memory usage.

Original OpenMP-

The general observation in case of Original OpenMP for both 32 and 64 bit is that execution of the benchmark in terms of MFLOPS performance with OpenMP in general exceeds that without OpenMP, with a decrease in performance in both cases for an increase in the number of 4 byte words for a constant number of ops per word.

References

- http://www.roylongbottom.org.uk/linux%20openmp%20benchmarks.htm#anchorStart