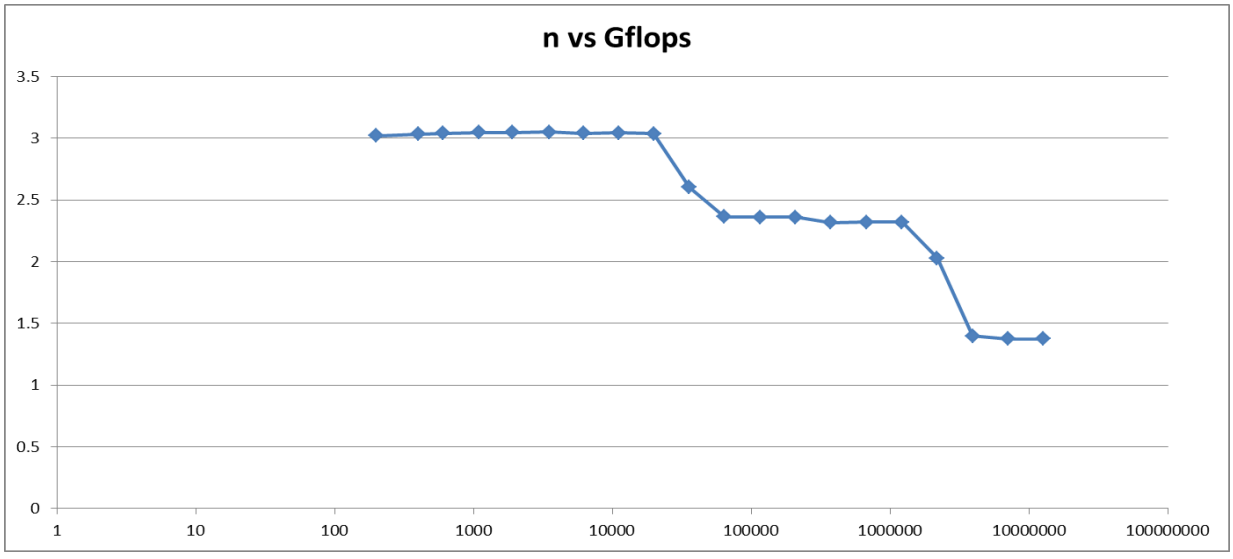
Lab assignment- 2

Part-a

K No.of Accumulators	Unrolling Factor L					
	2.598183	2.58409	2.583484	2.582772	2.581955	2.580259
	0	1.303627	0	1.302498	0	1.300568
	0	0	0.885801	0	0	0.885579
	0	0	0	0.886016	0	0
	0	0	0	0	0.88646	0
	0	0	0	0	0	0.887055

In the above table row represents the unrolling factor and column represents the number of accumulators used. When unrolling factor is 1 irrespective of the number of accumulators the performance remains constant. But as the unrolling factor increases the performance increases. The best performance is observed (after repeated executions) for unrolling factor-3 and when 3 accumulators are used. This program was run on stampede supercomputer with Intel Xeon E5 processor. The latency of the pipeline of add instruction is 3 and cycles per issue is 1. So the number of accumulators for best performance should be 3/1= 3. This is same our result.

Part – b



The above graph shows the required n vs Gflops plot, with n as x-axis and GFlops as y-axis. Clearly three plateaus can be seen. My first guess was cache misses. So I used the perf tool which confirmed

the guess. The (excerpts of) profiles with different values of n which the perf tool showed are given below:

PERF Results

```
Performance counter stats for './scalaradd analysis1' n=1:
25,048 L1-dcache-misses       #   0.01% of all L1-dcache hits
    719,424,452 cycles                #   0.000 GHz
  1,352,356,312 instructions          #   1.88  insns per cycle

Performance counter stats for './scalaradd analysis3'n=3:
    493,080,777 L1-dcache-loads
     27,832 L1-dcache-misses       #   0.01% of all L1-dcache hits
    913,877,049 cycles                #   0.000 GHz
  1,985,811,871 instructions          #   2.17  insns per cycle

Performance counter stats for './scalaradd analysis6' n=6:
    359,150,291 L1-dcache-loads
     67,707 L1-dcache-misses       #   0.02% of all L1-dcache hits
    616,338,053 cycles                #   0.000 GHz
  1,438,160,228 instructions          #   2.33  insns per cycle
      2,634 cache-misses
Performance counter stats for './scalaradd analysis8' n=8:
    569,137,494 L1-dcache-loads
     71,266,042 L1-dcache-misses       #   12.52% of all L1-dcache hits
    966,309,297 cycles                #   0.000 GHz
  2,276,490,036 instructions          #   2.36  insns per cycle
      2,575 cache-misses

Performance counter stats for './scalaradd analysis10' n=10':
    459,006,583 L1-dcache-loads
     57,377,433 L1-dcache-misses       #   12.50% of all L1-dcache hits
    847,148,817 cycles                #   0.000 GHz
  1,834,271,342 instructions          #   2.17  insns per cycle
```

```
Performance counter stats for './scalaradd analysis13'n=13:
    667,712,707 L1-dcache-loads
     83,431,507 L1-dcache-misses       #   12.50% of all L1-dcache hits
  1,320,705,193 cycles                #   0.000 GHz
  2,660,929,382 instructions          #   2.01  insns per cycle

Performance counter stats for './scalaradd analysis16':
    488,269,303 L1-dcache-loads
     61,104,221 L1-dcache-misses       #   12.51% of all L1-dcache hits
    970,233,344 cycles                #   0.000 GHz
  1,897,419,324 instructions          #   1.96  insns per cycle
     92,596 cache-misses

Performance counter stats for './scalaradd analysis17' n=17:
    441,077,524 L1-dcache-loads
     55,184,553 L1-dcache-misses       #   12.51% of all L1-dcache hits
    983,719,219 cycles                #   0.000 GHz
  1,664,494,040 instructions          #   1.69  insns per cycle

Performance counter stats for './scalaradd analysis18' n=1':
    399,976,052 L1-dcache-loads
     49,968,098 L1-dcache-misses       #   12.49% of all L1-dcache hits
  1,117,325,224 cycles                #   0.000 GHz
  1,420,564,500 instructions          #   1.27  insns per cycle
     13,609,834 cache-misses

Performance counter stats for './scalaradd analysis20' n=20:
    656,960,185 L1-dcache-loads
     81,950,823 L1-dcache-misses       #   12.47% of all L1-dcache hits
  1,822,359,886 cycles                #   0.000 GHz
  2,047,631,503 instructions          #   1.12  insns per cycle
```

As seen in the above results the number of data cache misses increased with increase of n. This program was run on stampede supercomputer with Intel Xeon E5 processor. This has line size of 64 bytes and 32 kb L1 dcache an 32kb L1 icache. Thus after n=9 the data goes out of L1 dcache. This explains data miss increase. For confirming this argument perf tool was used and the L1 dcache miss rate increased from n=6 to n=8. Now after n=17 even though the L1 dcache miss rate did not change much, the ipc decreased. This could be because of L2 misses. So when perf was used to find the number of cache misses , though the L1 cache miss rate did not change much the number of cache misses increased steeply from n=16 to n=18.