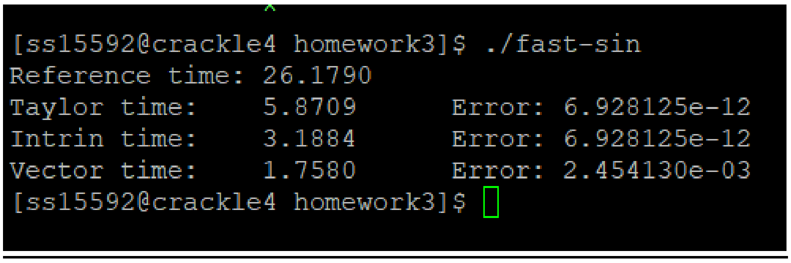
Question – 1: Pitch your final project

The email has already been sent with the detailed proposal of the project.

Question – 2: Approximating Special Functions Using Taylor Series & Vectorization

I have implemented the "Intrin time" function and the error is shown below. I have done this for the SSE part of the function $sin4\ intrin()$ and used crackle4 server of NYU (screenshot shown below).



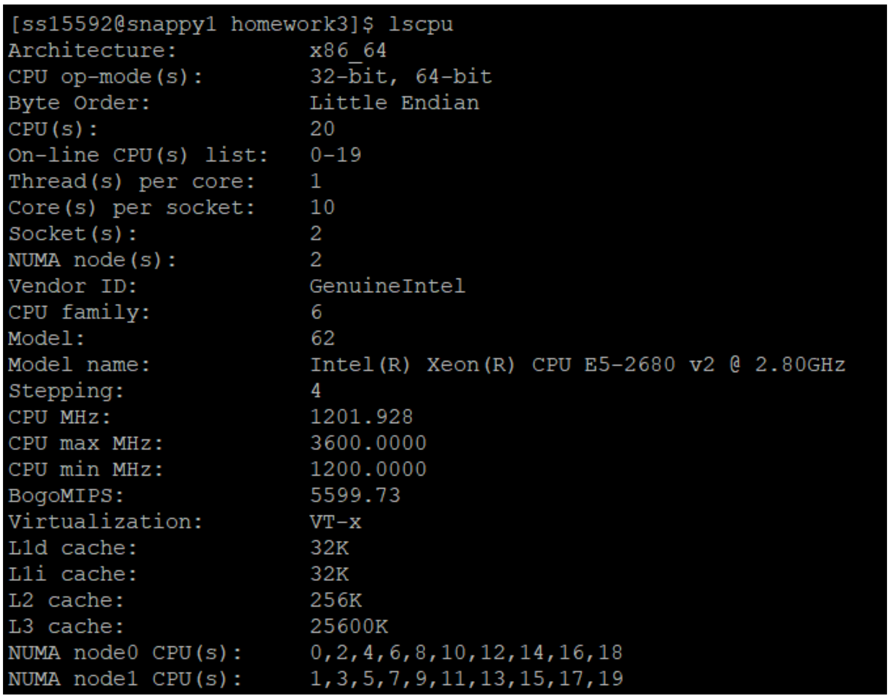
```
[ss15592@crackle4 homework3]$ ./fast-sin
Reference time: 26.1790
Taylor time:    5.8709      Error: 6.928125e-12
Intrin time:    3.1884      Error: 6.928125e-12
Vector time:    1.7580      Error: 2.454130e-03
[ss15592@crackle4 homework3]$
```

Extra Credit:

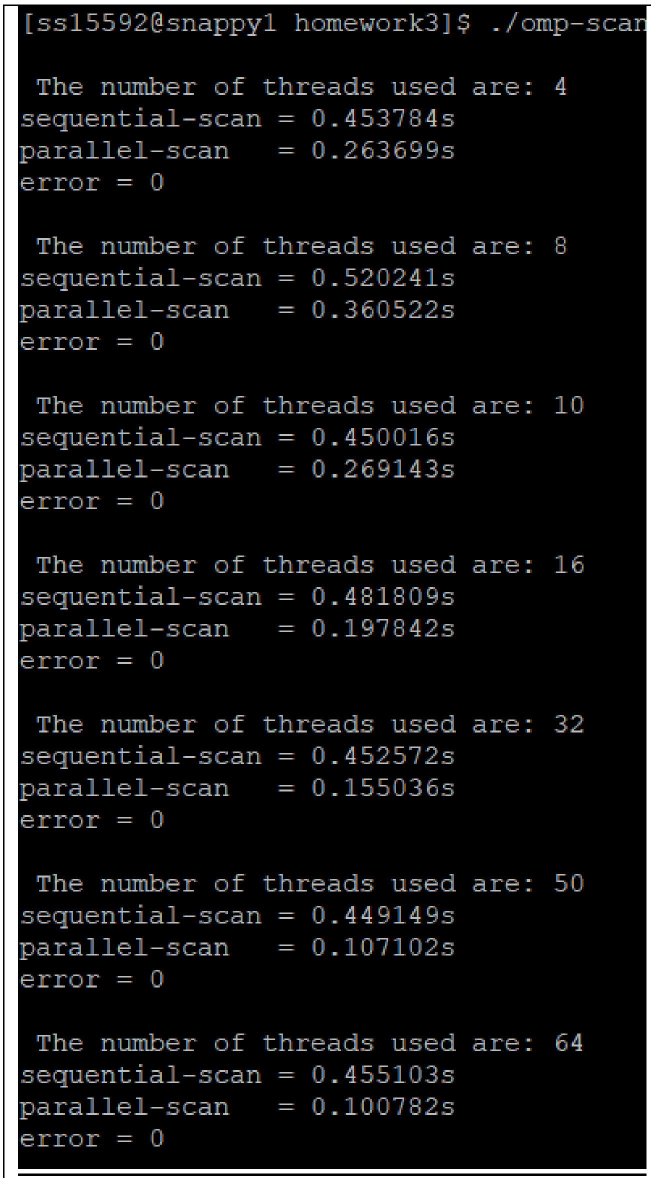
The key idea is to use the given identities to find out whether the sin series should be used or the cosine series to be used to evaluate $sin\theta$ due to the cyclic property of sine and cosine. A flag has been created to keep a note on the sign of the sum of the series.

Question – 3: Parallel scan with OpenMP

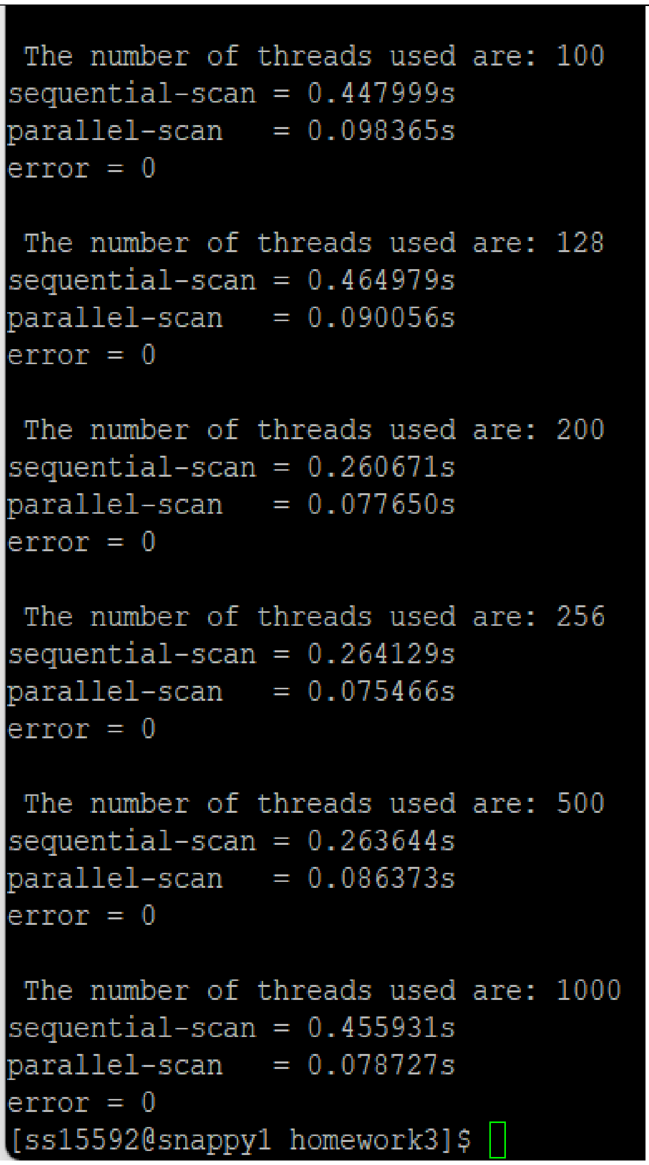
The run-times along with the error for different number of threads are shown in the adjacent image. The thread sizes are also shown in the image. The maximum speedup was 5.83. The snappy1 NYU server was used with Two Intel Xeon E5-2680 (2.80 GHz) (20 cores) and 128GB and Cent OS 7. The CPU architecture used is shown below in another screenshot. The number of cores on the CPU are 20. The runtime plots are attached on the next page.



```
[ss15592@snappy1 homework3]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                20
On-line CPU(s) list:   0-19
Thread(s) per core:    1
Core(s) per socket:    10
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 62
Model name:            Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz
Stepping:              4
CPU MHz:               1201.928
CPU max MHz:           3600.0000
CPU min MHz:           1200.0000
BogoMIPS:              5599.73
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              25600K
NUMA node0 CPU(s):     0,2,4,6,8,10,12,14,16,18
NUMA node1 CPU(s):     1,3,5,7,9,11,13,15,17,19
```



```
[ss15592@snappy1 homework3]$ ./omp-scan

 The number of threads used are: 4
sequential-scan = 0.453784s
parallel-scan   = 0.263699s
error = 0

 The number of threads used are: 8
sequential-scan = 0.520241s
parallel-scan   = 0.360522s
error = 0

 The number of threads used are: 10
sequential-scan = 0.450016s
parallel-scan   = 0.269143s
error = 0

 The number of threads used are: 16
sequential-scan = 0.481809s
parallel-scan   = 0.197842s
error = 0

 The number of threads used are: 32
sequential-scan = 0.452572s
parallel-scan   = 0.155036s
error = 0

 The number of threads used are: 50
sequential-scan = 0.449149s
parallel-scan   = 0.107102s
error = 0

 The number of threads used are: 64
sequential-scan = 0.455103s
parallel-scan   = 0.100782s
error = 0
```



```
 The number of threads used are: 100
sequential-scan = 0.447999s
parallel-scan   = 0.098365s
error = 0

 The number of threads used are: 128
sequential-scan = 0.464979s
parallel-scan   = 0.090056s
error = 0

 The number of threads used are: 200
sequential-scan = 0.260671s
parallel-scan   = 0.077650s
error = 0

 The number of threads used are: 256
sequential-scan = 0.264129s
parallel-scan   = 0.075466s
error = 0

 The number of threads used are: 500
sequential-scan = 0.263644s
parallel-scan   = 0.086373s
error = 0

 The number of threads used are: 1000
sequential-scan = 0.455931s
parallel-scan   = 0.078727s
error = 0
[ss15592@snappy1 homework3]$
```