# COMP-40730 HPC

# REPORT FOR ASSIGNMENT 2


| | |
|---|---|
| Author: | Paula Dwan |
| Due date: | July-2014 |
| Lecturer: | Alexey Lastovetsky |
| Subject: | COMP-40730 High Performance Computing |
| College: | University College Dublin |

## CONTENTS


**Contents** ..... **2**
**Exercise** ..... **3**
**Overview of Computations Obtained and How** ..... **4**
Assignment Execution ..... 6
Running A2-pthreads-manual-<atlas | cblas> : standalone ..... 7
Running A2-pthreads-solo : standalone ..... 8
Log Files Obtained ..... 8
GNUplot Execution ..... 9
Matrix Sizes Evaluated ..... 10
Row-Major as applied ..... 10
**Summary Results** ..... **11**
GNUplot Graphs – matrix range 1 : 50 → 100 : <increment> ..... 11
*$ ./runAssignment2.sh -i -v : manual – sample run ..... 11*
*$ ./runAssignment2.sh -i -v : solo | pThreads – sample run ..... 12*
GNUPlot Graphs – Matrix Range 2 : 50 → 1000 : <random 1 → 10> ..... 13
*$ ./runAssignment2.sh -r -v : manual – sample run ..... 13*
*$ ./runAssignment2.sh -r -v : solo | pThreads – sample run ..... 14*
GNUplot Graphs – matrix range 3 : 50 → 1000 : <increment> ..... 15
*$ ./runAssignment2.sh -i -v : manual – sample run ..... 15*
*$ ./runAssignment2.sh -i -v : solo | pThreads – sample run ..... 16*
Conclusions ..... 16
**Appendices** ..... **17**
Appendix I – Validate Results ..... 17
*A2-pthreads-manual-cblas ..... 17*
*A2-pthreads-solo ..... 19*
Appendix II – Summary of Results Obtained ..... 21
*./A2-pthreads-manual-cblas -r 50 → 100 : range 1 ..... 21*
*./A2-pthreads-manual-cblas -i 50 → 100 : range 1 ..... 21*
*./A2-pthreads-manual-cblas -r 50 → 1000 : range 2 ..... 22*
*./A2-pthreads-manual-cblas -i 50 → 1000 : range 2 ..... 22*
*./A2-pthreads-manual-cblas -r 50 → 1000 : range 3 ..... 23*
*./A2-pthreads-manual-cblas -i 50 → 1000 : range 3 ..... 23*
*./A2-pthreads-solo -i 50 → 100 : range 1 ..... 24*
*./A2-pthreads-solo -r 50 → 100 : range 1 ..... 24*
*./A2-pthreads-solo -i 50 → 1000 : range 2 ..... 25*
*./A2-pthreads-solo -r 50 → 1000 : range 2 ..... 25*
*./A2-pthreads-solo -i 50 → 1000 : range 3 ..... 26*
*./A2-pthreads-solo -r 50 → 1000 : range 3 ..... 26*
Appendix III – References / Acknowledgements ..... 27

## EXERCISE

Write a parallel Pthreads program computing the norm of the product of two n×n dense matrices on a p-processor SMP so that

- p threads are involved in the parallel computations.
- The 1-dimensional parallel algorithm of matrix multiplication is employed:
  - one of matrices is partitioned in one dimension into p equal slices
  - there is one-to-one mapping between the partitions and threads
  - each thread is responsible for computation of the corresponding slice of the resulting matrix
- Computation of the norm of the resulting matrix employs the mutex synchronization mechanism.

You can use BLAS or ATLAS for local computations.

Experiment with the program and build:

- The dependence of the execution time of the program on the matrix size n.
- The speedup over a serial counterpart of the program.

Explain the results.

---

**Variants of the assignment:**

1. Granularity of the program:
   - (a) Two successive steps:
     - i. Parallel matrix multiplication
     - ii. Parallel computation of the norm of the resulting matrix
   - (b) One-step algorithm. No intermediate resulting matrix.

2. Partitioning scheme:
   - (a) Left matrix is horizontally partitioned
   - (b) Right matrix is vertically partitioned

3. Matrix norm to be computed:
   - (a) The maximum absolute column sum norm (aka one-norm):

$$\|A\|_1 = \max_{0 \le j < n} \sum_{i=0}^{n-1} |a_{ij}|$$

   - (b) The maximum absolute row sum norm (aka infinity-norm):

$$\|A\|_\infty = \max_{0 \le i < n} \sum_{j=0}^{n-1} |a_{ij}|$$

## Overview of Computations Obtained and How

Assignment 2 basically involved (for me) writing two programmes,

- **A2-pthreads-manual.c**
  one which contained the code for manual straight-forward IJK computation and dgemm to calculate matrix |C| and the infinity norm of |C|.
- **A2-pthreads-solo.c**
  second which contained the code for manual straight-forward IJK computation and pThreads to calculate matrix |C| and the infinity norm of |C|.

I then compared the time taken to calculate matrix |C| for each method using [N]x[N] matrices |A| and |B| of different sizes, allocating values for the |A| and |B| using random numbers (1 to 10 inclusive) or the column value + 1 (so that column 1000 would have a value of 1001).

1. **manual straight-forward IJK computation**

   Implementation of a straight forward matrix nxn multiplication.

   *Present in **A2-pthreads-manual.c** and in **A2-pthreads-solo.c***

**Code**

```
for (ni=0 ; ni<rows ; ni++)
{
   for (nj=0 ; nj<cols ; nj++)
   {
      double sum = 0.0 ;
      for (nk=0 ; nk<rows ; nk++)
      {
         sum+= (A[(ni*rows)+nk]) * (B[(nk*rows)+nj]) ;
      }
      C[(ni*rows)+nj] = sum ;
   }
}
```

2. **Dgemm for straight-forward IJK computation**

   I used dgemm compiled for atlas or for cblas.

   The matrix |C| was calculated using cblas by default otherwise the user could decide to re-build and execute using cblas or atlas).

   *Present in **A2-pthreads-manual.c** only*

**Code**

```
   int ni, nj ;
// m, n, k : local integers indicating the size of the matrices for
// rows x columns :: A : m x  k, B : k x n, C: m x n
// Here, m = n = k = rows = columns = <nx> = <ny> as supplied
   int lm = rows, ln = rows ;
// la_offset, lb_offset, lc_offset :
// Leading dimension of matrix A, B or C respectively, or the number of elements
// between successive rows for row-major storage or columns for column-major
// storage.
   int la_offset = rows, lb_offset = cols, lc_offset = rows ;
   int ALPHA=1.0 ;
   int BETA=0.0 ;

   cblas_dgemm( CblasRowMajor, CblasNoTrans, CblasNoTrans, lm, ln, ln, ALPHA, \
      A, la_offset, B, lb_offset, BETA, C, lc_offset) ;
```

### 3. pThreads calculation

This was completed using a data structure, creating a new thread to correspond to the each slice. Infinity norm was mutes'ed to ensure only one thread could access it for updates at any one time.

*Present in **A2-pthreads-solo.c** only*

**Code**

```
// create separate slices for each thread
   for (ni = 0; ni < nt; ni++)
  {
      slice[ni].sliceB = allB + ni * num_per_slice;
      slice[ni].sliceC = allC + ni * num_per_slice;
      slice[ni].mutex = mutex;
      slice[ni].num_of_rows = (ni == nt - 1) ? nx-ni * num_per_slice : num_per_slice;

// create separate threads for each slice
      pthread_create(&working_thread[ni], NULL, pthread_matrix_slice_multiply, \
         (void *)&slice[ni]);
  }

// wait for each thread to terminate
   for (ni = 0; ni < nt; ni++)
   {
      pthread_join(working_thread[ni], NULL);
   }

// compute |C| → used dgemm
```

Thus results were obtained for each option 1., 2. and 3. using the same source |A| and |B| for both -i (increment) and also using -r (random) .

Multiple implementation of each .c program was enabled using ***./runAssignment2.sh,*** for example :

```
$ ./runAssignment2.sh –1 –i –v
```

This runs the c. program using cblas and manual straight-forward IJK for incremental column values, using predefined matrix sizes for each implementation.

Single implementation is completed using

- **./A2-pthreads-manual.c.**

  ```
  $ ./A2-pthreads-manual-cblas –i 10 file.txt file.dat
  ```

  This is compiled using *gcc* for *atlas* and *cblas,* as follows :

  ```
  gcc -I/home/cs/khasanov/libs/CBLAS/src A2-pthreads-manual.c -o A2-pthreads-manual-cblas \
       /home/cs/khasanov/libs/cblas_LINUX.a  /usr/lib/libblas.a -lgfortran
  gcc -o A2-pthreads-manual-atlas A2-pthreads-manual.c  \
       -I/home/cs/khasanov/libs/ATLAS/include/ \
       -L/home/cs/khasanov/libs/ATLAS/lib/Linux_UNKNOWNSSE2_4/ -lcblas -latlas -lm -O3
  ```

- **./A2-pthreads-solo.c.**

  ```
  $ ./A2-pthreads-solo –i 10 2 file-solo.txt file-solo.dat
  ```

  This is compiled using *gcc* for *pthread,* as follows :

  ```
  cc -Wall -I/home/cs/khasanov/libs/CBLAS/src -o A2-pthreads-solo  A2-pthreads-solo.c \
       /home/cs/khasanov/libs/cblas_LINUX.a  /usr/lib/libblas.a -lgfortran -pthread
  ```

### *ASSIGNMENT EXECUTION*

The compiled .c program ./A2-pthreads-manual-<cblas|atlas> was executed multiple times standalone or using the script *./runAssignment2.sh* to obtain as wide a range of time taken to calculate |C| using each algorithm.

This has multiple options and the syntax and usage follows :

```
pdwan@csserver.ucd.ie
File  Edit  View  Search  Terminal  Help
[pdwan@csserver Assignment2]$ ./runAssignment2.sh

USAGE : ./runAssignment2.sh \
        -1|--manual -2|--solo -d1|--atlas -d2|--cblas -r|--random -i|--increment -m|--matrix <n> -t|--thread <t> -v|--values -?|-h|--help

TO :    Calculate |C| = |A| x |B| and then infintiy norm using pthreads

LOGS :  Created in current dir and moved to [ logDir ] :
        <file>.txt :    matrix values for matrices |A| |B| & |C|
        <file>.dat :    timing data for each computation
        <file>.log :    summary of stdout.

WHERE : -1|--manual     Compile A2-pthreads-manual.c : straight-forward IJK and DGEMM computations only
        -2|--solo       Compile A2-pthreads-solo.c   : straight-forward IJK and pThreads computations only, only valid on 'yeats.ucd.ie'
                        '-1|--manual' and '-2|--solo' are mutually exclusive

        -d1|--atlas     Compile  A2-pthreads-manual.c source files using dgemm atlas
        -d2|--cblas     Compile  A2-pthreads-manual.c source files using dgemm cblas

        -r|--random     Initialize |A| & |B| with random numbers and |C| with '0'
        -i|--increment  Initialize |A| & |B| incrementally with <column> value and |C| with '0'
                        '-i|--increment' & '-r|--random' are mutually exclusive

        -m|--matrix <n> Matrix dimension, if odd number +1 added or if invalid set to [ 1000 ], thread count set to [ 100 ]
        -t|--thread <t> number of threads, if invalid set to  [ 100 ]  and matrix size set to [ 1000 ]
        -v|--values     Use predefined range of valid values for <nx> and for <np> as follows :
                        Range 1 <matrixArray> : { 50 50 50 100 100 100 500 500 500 1000 1000 1000 }
                                <threadArray> : { 10 10 10 10 10 10 20 20 20 20 20 20 20 20 }
                        Range 2 <matrixArray> : { 50 50 50 50 50 50 100 100 100 100 100 100 }
                                <threadArray> : { 2 2 2 5 5 5 10 10 10 20 20 20 }
                        Range 3 <matrixArray> : { 50 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850 900 950 1000 }
                                <threadArray> : { 10 10 20 20 30 30 40 40 50 50 60 60 70 70 80 80 90 90 100 }
                        '-m|--matrix <n>' & '-t|--thread <t>' are mutually exclusive of  '-v|--values'.

        -?|-h|--help    usage

[pdwan@csserver Assignment2]$
```

Execute this script in the home directory of Assignment 2.

Sample execution follows for range 3 :

```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

```
$ ./runAssignment2.sh -1 -d2 -r -v
```

```
[pdwan@csserver Assignment2]$ ./runAssignment2.sh -1 -d2 -r -v

# RUNNING :     ./A2-pthreads-manual-cblas -r 50
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <50> x <50> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.001198] seconds and has infinity norm of [89502.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.000277] seconds and has infinity norm of [89502.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               50              0.001198        89502.0                 0.000277        89502.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-manual-cblas -r 150
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <150> x <150> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.031242] seconds and has infinity norm of [772112.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.006532] seconds and has infinity norm of [772112.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               150             0.031242        772112.0                0.006532        772112.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-manual-cblas -r 200
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <200> x <200> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.094799] seconds and has infinity norm of [1322244.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.016427] seconds and has infinity norm of [1322244.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               200             0.094799        1322244.0               0.016427        1322244.0
# CLEAN-UP ...
```

Sample execution follows for range 1 for pThreads:

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

```
$ ./runAssignment2.sh –2 –r –v
```

```
pdwan@yeats.ucd.ie
File  Edit  View  Search  Terminal  Help
-bash-3.2$
-bash-3.2$
-bash-3.2$ ./runAssignment2.sh -2 -r -v

# RUNNING :     ./A2-pthreads-solo -r 50 2
# ALLOCATE :    matrices |allA|, |allB| ...
# INITIALIZE :  |allA| & |allB| ...
# INITIALIZE :  <50> x <50> matrix |allC| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |allC| calculated in [0.001412] seconds and has infinity norm of [89502.0] ...
# INITIALIZE :  |allC| for pThreads computation ...
# RESULTS :     pThreads computation ...
#               Matrix |allC| calculated in [0.000808] seconds and has infinity norm of [85625.0] ...
# SUMMARY :     |Matrix|        |Threads|       Time/manual     Inf Norm/manual         Time/pThreads   Inf Norm/pThreads
#               50              2               0.001412        89502.0                 0.000808        85625.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-solo -r 50 2
# ALLOCATE :    matrices |allA|, |allB| ...
# INITIALIZE :  |allA| & |allB| ...
# INITIALIZE :  <50> x <50> matrix |allC| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |allC| calculated in [0.001403] seconds and has infinity norm of [89502.0] ...
# INITIALIZE :  |allC| for pThreads computation ...
# RESULTS :     pThreads computation ...
#               Matrix |allC| calculated in [0.000715] seconds and has infinity norm of [85625.0] ...
# SUMMARY :     |Matrix|        |Threads|       Time/manual     Inf Norm/manual         Time/pThreads   Inf Norm/pThreads
#               50              2               0.001403        89502.0                 0.000715        85625.0
# CLEAN-UP ...

# RUNNING :     ./A2-pthreads-solo -r 50 2
# ALLOCATE :    matrices |allA|, |allB| ...
# INITIALIZE :  |allA| & |allB| ...
# INITIALIZE :  <50> x <50> matrix |allC| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |allC| calculated in [0.001402] seconds and has infinity norm of [89502.0] ...
# INITIALIZE :  |allC| for pThreads computation ...
# RESULTS :     pThreads computation ...
#               Matrix |allC| calculated in [0.000763] seconds and has infinity norm of [85625.0] ...
# SUMMARY :     |Matrix|        |Threads|       Time/manual     Inf Norm/manual         Time/pThreads   Inf Norm/pThreads
#               50              2               0.001402        89502.0                 0.000763        85625.0
# CLEAN-UP ...
```

Please retain the overall directory structure when unzipping.

Note that the script *./runAssignment2.sh* allows two types of implementation

- Multiple iteration
  use the switch <-v | --values>, when a predefined range applies for [N] : matrix size and [T] : Thread size.
- Single iteration
  use the switch <-m | --matrix> [N] where the user specifies value for [N] : matrix size
  and the switch <-t | --thread> [T] where the user specifies value for [T] : Thread size.

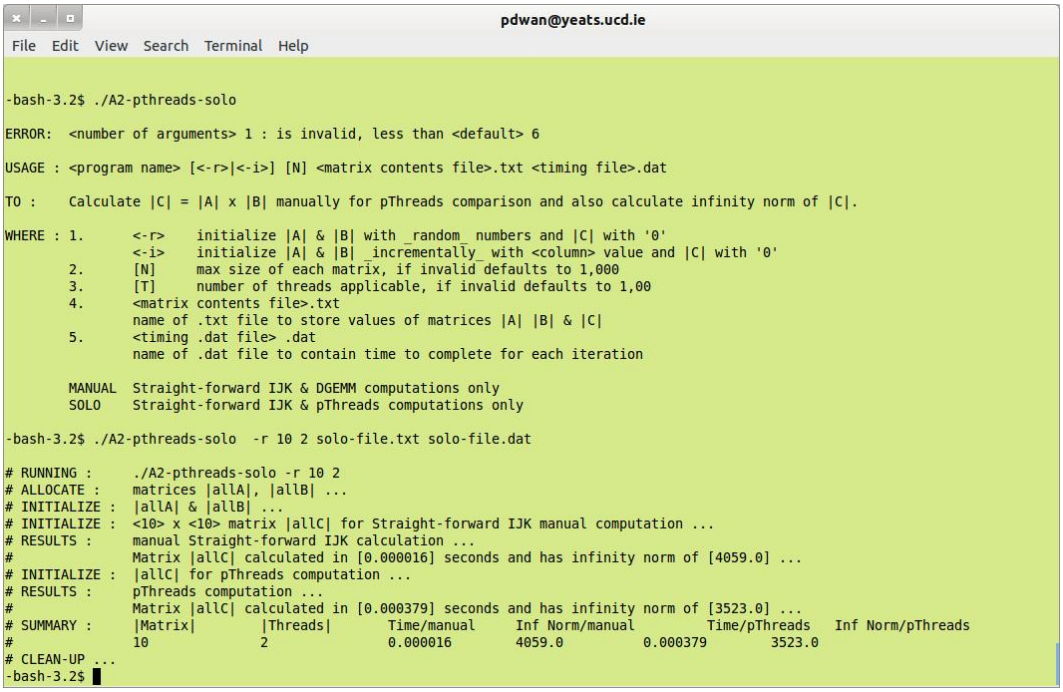
### *RUNNING* A2-PTHREADS-MANUAL-<ATLAS | CBLAS> *: STANDALONE*

The compiled .c program may also be run standalone. Usage and sample execution follows :

```
pdwan@csserver.ucd.ie
File  Edit  View  Search  Terminal  Help
[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas

ERROR:  <number of arguments> 1 : is invalid, less than <default> 5

USAGE : <program name> [<-r>|<-i>] [N] <matrix contents file>.txt <timing file>.dat

TO :    Calculate |C| = |A| x |B| manually for pThreads comparison and also calculate infinity norm of |C|.

WHERE : 1.      <-r>    initialize |A| & |B| with _random_ numbers and |C| with '0'
                <-i>    initialize |A| & |B| _incrementally_ with <column> value and |C| with '0'
        2.      [N]     max size of each matrix, if invalid defaults to 1,000
        3.      <matrix contents file>.txt
                name of .txt file to store values of matrices |A| |B| & |C|
        4.      <timing .dat file> .dat
                name of .dat file to contain time to complete for each iteration

        MANUAL  Straight-forward IJK & DGEMM computations only
        SOLO    Straight-forward IJK & pThreads computations only

[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas -r 10 manual.11.txt manual.11.dat

# RUNNING :     ./A2-pthreads-manual-cblas -r 10
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <10> x <10> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.000019] seconds and has infinity norm of [4059.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.000010] seconds and has infinity norm of [4059.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               10              0.000019        4059.0          0.000010        4059.0
# CLEAN-UP ...
[pdwan@csserver Assignment2]$
```

### *RUNNING* A2-PTHREADS-SOLO *: STANDALONE*

The compiled .c program may also be run standalone. Usage and sample execution follows :




### *LOG FILES OBTAINED*

Data text files suitable containing the values of the computation used for matrices |A| and |B| and the results stored in |C| are saved in the appropriate log files. File naming convention via the script is :

| | |
|---|---|
| <data log file name> | `Values-<time>-A2-pthreads-<iteration>.txt` |
| example: | `Values-20140715.170928-A2-pthreads-0.txt`<br>`file-10txt` |

Single iteration also applies where the user enters arbitrary, valid values for matrix size and does not use the scripts and the other required parameters. Each new matrices |A| and |B| and the results in |C| were saved to the data file, thus simple validation using *LibreOffice Calc*.

A summary file containing processing time for each computation (manual & DGEMM and manual & pThreads) is also saved. This is in a format suitable for us with GNUplot.

| | |
|---|---|
| <timing log file name> | `Data-<time>-A2-pthreads-1D.dat` |
| example: | `Data-20140715.171337-A2-pthreads-1D.dat`<br>`Data-20140715.172124-A2-pthreads-1D.dat`<br>`file-10dat` |

I did not save a separate .dat file for each run of the script for each algorithm. Instead each .dat file contains the time taken for each matrix size for the preset range of values. **./runAssignment2.sh** may be updated with more if needed but the following are those in use at the moment.

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
#  Matrix & pThreads - range 2
   declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
   declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

For compilation using the script, a suffix of ***-atlas*** indicates compilation for atlas and a suffix of ***-cblas*** indicates that the c program was compiled via cblas. No suffix indicates compilation using -***pthread*** for pThreads.

Finally a log file containing a listing of each algorithm used for that iteration.

After each run, all .log, .txt, .dat and .bup files are copied to the directory *logDir/*.

If either compiled file is used without the script then .dat and .txt files may be named whatever the user wishes and no .log file applies.

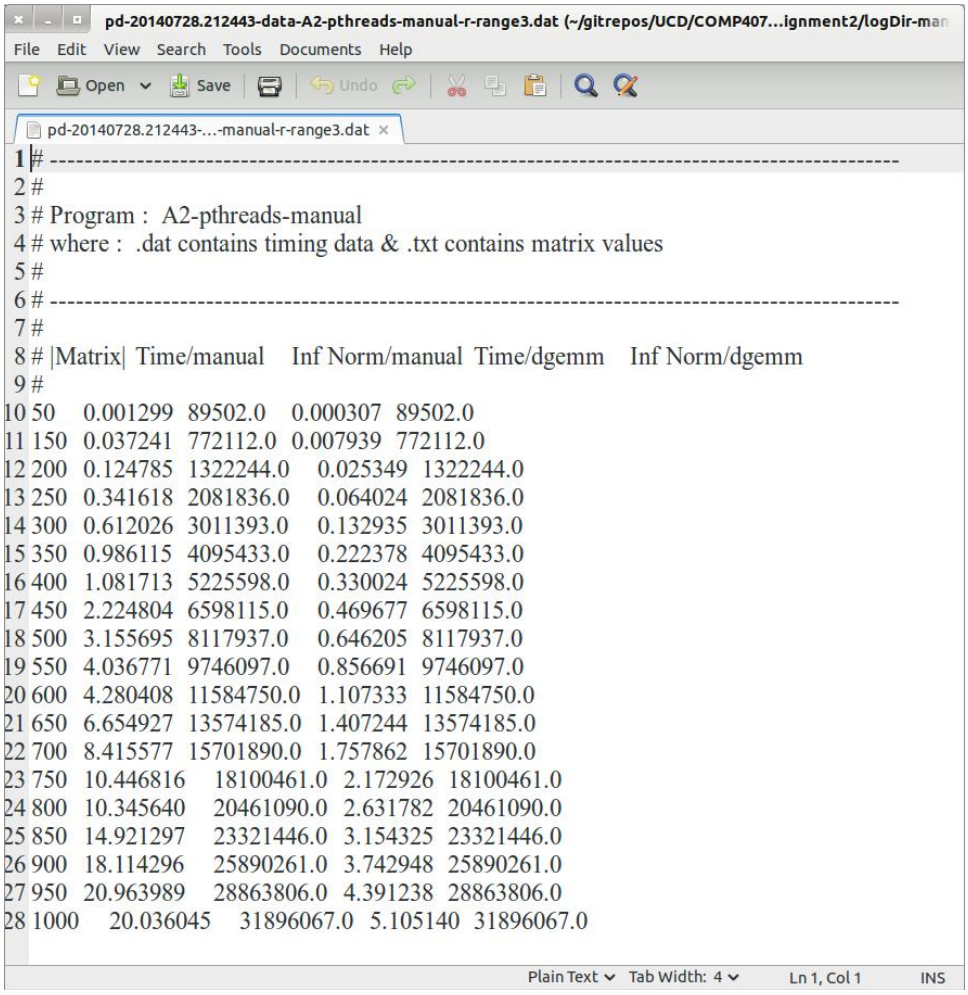
I wished to keep each .c program as clean as possible and so all production setup was completed in the script for each assignment. Thus file creation and validation for each iteration was completed before the .c program was even called. Simple validation of the arguments passed to each .c program is also completed if ran standalone.

I also spot-checked the results as practical. Results spot-check are detailed in Appendix I – Validate Results.

### *GNUPLOT EXECUTION*

I followed the same structure for each .dat file as produced, an example follows :

**Sample .dat file**




If wished, the .txt file contains the matrices |A| and |B| used to calculate |C| and the type of computation applicable and the time taken to complete. The .dat file is just a summary of the matrix sizes (when the later is applicable) as well as time taken for each type of computation.

The contents of each .dat was then presented in graphical format using GNUplot, comparing times taken for manual and for BLAS/ATLAS computations.

**Sample GNUplot program execution**

```
# To execute, launch GNUplot and run :
# gnuplot> load <filename.gp>
# making sure that the data file name used is updated if needed.
# ------------------------------------------------------------------------
# Paula Dwan : Assignment 2
reset
set xrange [0:100]
set xtic (0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
set ytic auto
set size 1,1
set grid
set key outside
#
set title 'pThread Comparison : Matrix Size -v- Time Taken'
set ylabel 'Time taken / seconds'
set xlabel 'Matrix size / NxN'
set origin 0,0
plot 'logDir/manual-100.dat' u 1:2 t 'manual' w l lw 0.5 lc rgb 'blue',
'logDir/manual-100.dat' u 1:4 t 'dgemm' w l lw 0.5 lc rgb 'red'
#
pause -1
```

Thankfully for Linux (Ubuntu) – I could install and run GNUplot locally.

Screen shots of each were taken and added to the section Summary Results.

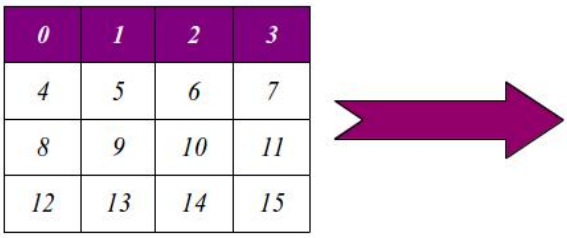
### *Matrix Sizes Evaluated*

When applying the three options I used matrices |A| and |B| of varying sizes from [ 10x10 ] to [ 1000x1000 ]. The values in each matrix were dependent on the switch

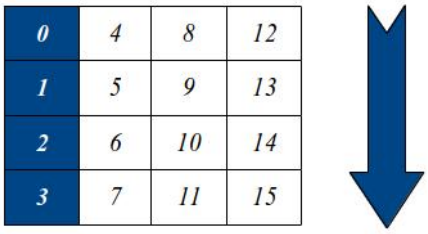
- **-r**
  random from 1 to 10, so each cell regardless of matrix size had a value of 1 to 10 inclusive. This reduced computation time based on large cell values.
- **-i**
  increment based on column index + 1, so all cells in column 1000 had a value of 1001. This could increase computation time of larger matrices as the cell would have comparatively larger values also.

### *Row-Major as applied*

For the straight-forward algorithm of **|C|**$_{[ij]}$ += **|A|**$_{[ik]}$ * **|B|**$_{[kj]}$ uses row major as follows (e.g.: [ 4x4 ] matrix:




Whereas column-major is as follows :



## SUMMARY RESULTS

| | |
|---|---|
| **Build/plot:** | • The dependence of the execution time of the program on the matrix size n.<br>• The speedup over a serial counterpart of the program. |
| ***Variant :*** | • *One-step algorithm. No intermediate resulting matrix.*<br>• *Left matrix is horizontally partitioned*<br>• *The maximum absolute row sum norm (aka infinity-norm):*<br>$\|A\|_\infty = \max_{0 \le i < n} \sum_{j=0}^{n-1} \|a_{ij}\|$ |
| ***Infinity norm*** | *Sum the absolute values along each row and then take the largest value as the answer.*<br>*Example:* $A = \begin{vmatrix} 1 & -7 \\ -2 & -3 \end{vmatrix}$<br>*then matrix norm of A = max ( 1 + \|-7\|, \|-2\| + \|-3\| ) = max ( 8, 5 ) =* ***8*** |

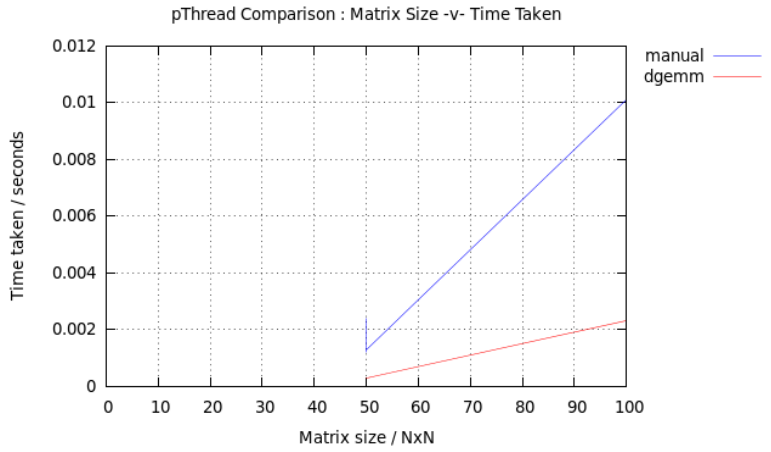
### *GNUPLOT GRAPHS – MATRIX RANGE 1 : 50 → 100 : <INCREMENT>*

***$ ./runAssignment2.sh -i -v : manual – sample run***

**Data**

| \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|
| **50** | 0.001259 | 1,625,625.0 | 0.000283 | 1,625,625.0 |
| **50** | 0.002391 | 1,625,625.0 | 0.000295 | 1,625,625.0 |
| **50** | 0.001297 | 1,625,625.0 | 0.000287 | 1,625,625.0 |
| **50** | 0.001232 | 1,625,625.0 | 0.000280 | 1,625,625.0 |
| **50** | 0.001216 | 1,625,625.0 | 0.000291 | 1,625,625.0 |
| **50** | 0.001267 | 1,625,625.0 | 0.000289 | 1,625,625.0 |
| **100** | 0.010098 | 25,502,500.0 | 0.002294 | 25,502,500.0 |
| **100** | 0.010646 | 25,502,500.0 | 0.002014 | 25,502,500.0 |
| **100** | 0.009306 | 25,502,500.0 | 0.004446 | 25,502,500.0 |
| **100** | 0.009266 | 25,502,500.0 | 0.001982 | 25,502,500.0 |
| **100** | 0.009307 | 25,502,500.0 | 0.001979 | 25,502,500.0 |
| **100** | 0.009287 | 25,502,500.0 | 0.001974 | 25,502,500.0 |

**Graph**

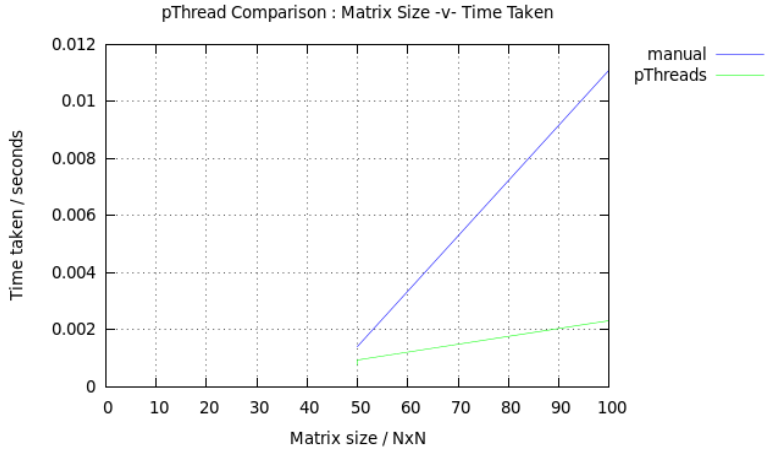


*$ ./runAssignment2.sh -i -v : solo | pThreads – sample run*

**Data**

| \|Matrix\| | \|Threads\| | Time / Manual | Inf Norm / manual | Time / pThreads | Inf Norm / pThreads |
|---|---|---|---|---|---|
| **50** | 2 | 0.001402 | 1,625,625.0 | 0.000780 | 1,625,625.0 |
| **50** | 2 | 0.001404 | 1,625,625.0 | 0.000777 | 1,625,625.0 |
| **50** | 2 | 0.001415 | 1,625,625.0 | 0.000775 | 1,625,625.0 |
| **50** | 5 | 0.001408 | 1,625,625.0 | 0.000938 | 1,625,625.0 |
| **50** | 5 | 0.001405 | 1,625,625.0 | 0.000895 | 1,625,625.0 |
| **50** | 5 | 0.001412 | 1,625,625.0 | 0.000936 | 1,625,625.0 |
| **100** | 10 | 0.011072 | 25,502,500.0 | 0.002305 | 25,502,500.0 |
| **100** | 10 | 0.011012 | 25,502,500.0 | 0.002738 | 25,502,500.0 |
| **100** | 10 | 0.011017 | 25,502,500.0 | 0.002255 | 25,502,500.0 |
| **100** | 20 | 0.011010 | 25,502,500.0 | 0.003405 | 25,502,500.0 |
| **100** | 20 | 0.011088 | 25,502,500.0 | 0.003177 | 25,502,500.0 |
| **100** | 20 | 0.011006 | 25,502,500.0 | 0.003435 | 25,502,500.0 |

**Graph**



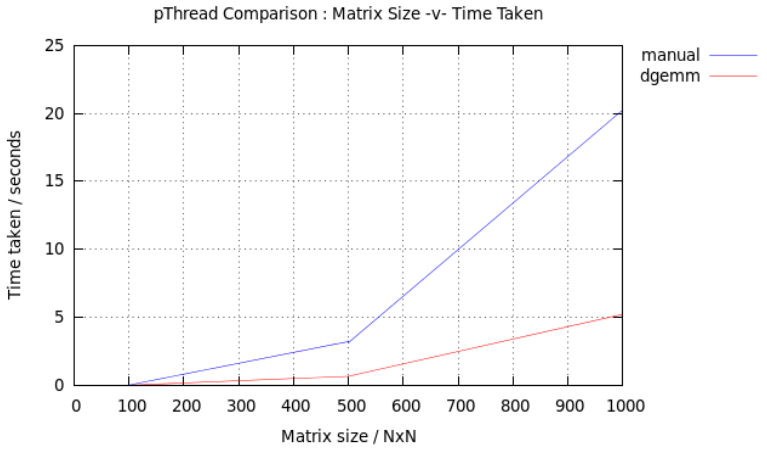
## *GNUPLOT GRAPHS – MATRIX RANGE 2 : 50 → 1000 : <RANDOM 1 → 10>*

### *$ ./runAssignment2.sh -r -v : manual – sample run*

**Data**

| \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|
| **50** | 0.001199 | 89,502.0 | 0.000275 | 89,502.0 |
| **50** | 0.001742 | 89,502.0 | 0.000279 | 89,502.0 |
| **50** | 0.001195 | 89,502.0 | 0.000278 | 89,502.0 |
| **100** | 0.009499 | 353,266.0 | 0.001992 | 353,266.0 |
| **100** | 0.009315 | 353,266.0 | 0.002046 | 353,266.0 |
| **100** | 0.009298 | 353,266.0 | 0.002074 | 353,266.0 |
| **500** | 3.247450 | 8,117,937.0 | 0.653315 | 8,117,937.0 |
| **500** | 3.119148 | 8,117,937.0 | 0.651457 | 8,117,937.0 |
| **500** | 3.311281 | 8,117,937.0 | 0.651434 | 8,117,937.0 |
| **500** | 3.091983 | 8,117,937.0 | 0.652713 | 8,117,937.0 |
| **1000** | 20.251307 | 31,896,067.0 | 5.214984 | 31,896,067.0 |
| **1000** | 20.300947 | 31,896,067.0 | 5.213675 | 31,896,067.0 |
| **1000** | 20.192011 | 31,896,067.0 | 5.229068 | 31,896,067.0 |
| **1000** | 20.505856 | 31,896,067.0 | 5.217558 | 31,896,067.0 |

**Graph**

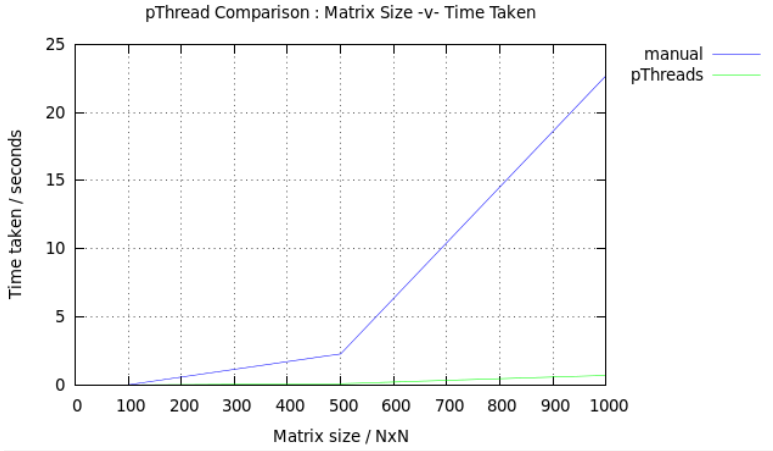


*$ ./runAssignment2.sh -r -v : solo | pThreads – sample run*

**Data**

| \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pThreads | Inf Norm / pThreads |
|---|---|---|---|---|---|
| **50** | 10 | 0.001409 | 89,502.0 | 0.001412 | 83,084.0 |
| **50** | 10 | 0.001408 | 89,502.0 | 0.001331 | 83,084.0 |
| **50** | 10 | 0.001413 | 89,502.0 | 0.001488 | 83,084.0 |
| **100** | 10 | 0.011018 | 353,266.0 | 0.003031 | 324,244.0 |
| **100** | 10 | 0.011017 | 353,266.0 | 0.002967 | 328,456.0 |
| **100** | 10 | 0.011027 | 353,266.0 | 0.003410 | 328,456.0 |
| **500** | 20 | 2.277439 | 8,117,937.0 | 0.100086 | 7,887,576.0 |
| **500** | 20 | 2.266145 | 8,117,937.0 | 0.079707 | 7,958,041.0 |
| **500** | 20 | 2.263969 | 8,117,937.0 | 0.072114 | 7,969,826.0 |
| **500** | 20 | 2.267955 | 8,117,937.0 | 0.089036 | 7,941,189.0 |
| **1000** | 20 | 22.695424 | 31,896,067.0 | 0.657843 | 31,359,528.0 |
| **1000** | 20 | 22.644039 | 31,896,067.0 | 0.845433 | 31,359,528.0 |
| **1000** | 20 | 22.586039 | 31,896,067.0 | 0.761873 | 31,359,528.0 |
| **1000** | 20 | 22.867197 | 31,896,067.0 | 0.689865 | 31,359,528.0 |

**Graph**



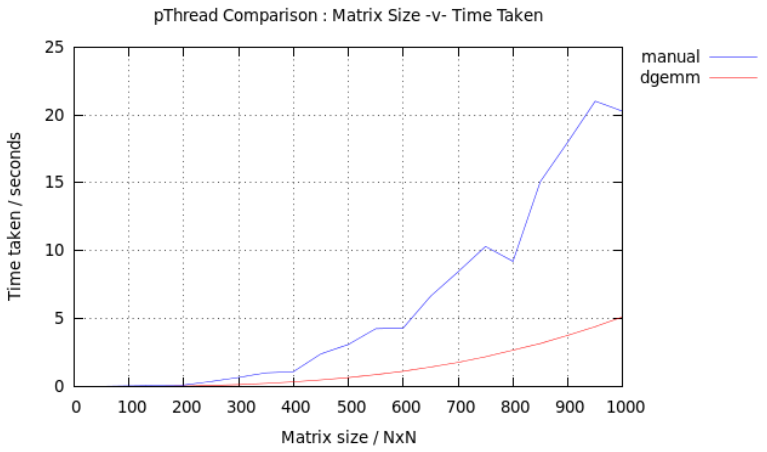
## *GNUPLOT GRAPHS – MATRIX RANGE 3 : 50 → 1000 : <INCREMENT>*

*$ ./runAssignment2.sh -i -v : manual – sample run*

**Data**

| \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|
| **50** | 0.001193 | 1,625,625.0 | 0.000272 | 1,625,625.0 |
| **150** | 0.062508 | 128,255,625.0 | 0.006597 | 128,255,625.0 |
| **200** | 0.093369 | 404,010,000.0 | 0.016680 | 404,010,000.0 |
| **250** | 0.352370 | 984,390,625.0 | 0.072505 | 984,390,625.0 |
| **300** | 0.652539 | 2,038,522,500.0 | 0.126024 | 2,038,522,500.0 |
| **350** | 0.993798 | 3,773,030,625.0 | 0.218930 | 3,773,030,625.0 |
| **400** | 1.081100 | 6,432,040,000.0 | 0.330587 | 6,432,040,000.0 |
| **450** | 2.387546 | 10,297,175,625.0 | 0.476119 | 10,297,175,625.0 |
| **500** | 3.082533 | 15,687,562,500.0 | 0.644611 | 15,687,562,500.0 |
| **550** | 4.249742 | 22,959,825,625.0 | 0.858860 | 22,959,825,625.0 |
| **600** | 4.298823 | 32,508,090,000.0 | 1.112652 | 32,508,090,000.0 |
| **650** | 6.642814 | 44,763,980,625.0 | 1.421403 | 44,763,980,625.0 |
| **700** | 8.440308 | 60,196,622,500.0 | 1.762349 | 60,196,622,500.0 |
| **750** | 10.307231 | 79,312,640,625.0 | 2.180169 | 79,312,640,625.0 |
| **800** | 9.210082 | 102,656,160,000.0 | 2.662343 | 102,656,160,000.0 |
| **850** | 15.091394 | 130,808,805,625.0 | 3.157122 | 130,808,805,625.0 |
| **900** | 18.004254 | 164,389,702,500.0 | 3.754617 | 164,389,702,500.0 |
| **950** | 21.020247 | 204,055,475,625.0 | 4.394798 | 204,055,475,625.0 |
| **1000** | 20.269382 | 250,500,250,000.0 | 5.117270 | 250,500,250,000.0 |

**Graph**




e

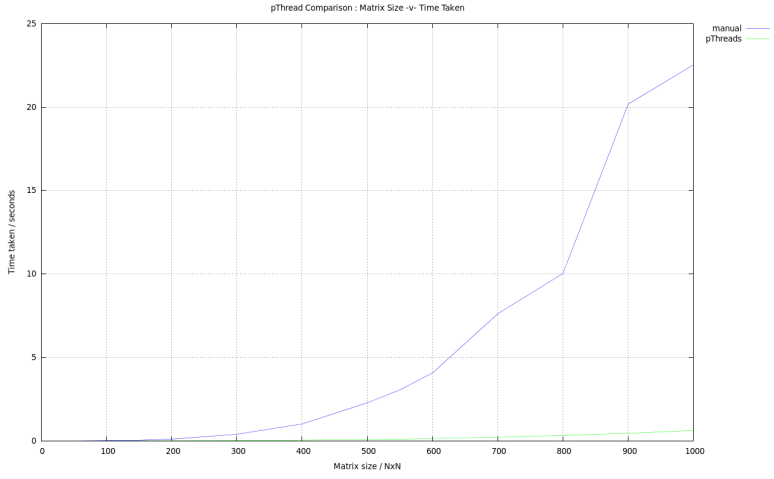
*$ ./runAssignment2.sh -i -v : solo | pThreads – sample run*

**Data**

| \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pThreads | Inf Norm / pThreads |
|---|---|---|---|---|---|
| **50** | 10 | 0.001402 | 1,625,625.0 | 0.001368 | 1,625,625.0 |
| **150** | 10 | 0.036497 | 128,255,625.0 | 0.005516 | 128,255,625.0 |
| **200** | 20 | 0.103509 | 404,010,000.0 | 0.010126 | 404,010,000.0 |
| **1000** | 100 | 22.578228 | 250,500,250,000.0 | 0.629005 | 250,500,250,000.0 |
| **300** | 30 | 0.394802 | 2,038,522,500.0 | 0.019049 | 2,038,522,500.0 |
| **1000** | 100 | 22.531387 | 250,500,250,000.0 | 0.623278 | 250,500,250,000.0 |
| **400** | 40 | 1.014198 | 6,432,040,000.0 | 0.045393 | 6,432,040,000.0 |
| **1000** | 100 | 22.699620 | 250,500,250,000.0 | 0.637117 | 250,500,250,000.0 |
| **500** | 50 | 2.278428 | 15,687,562,500.0 | 0.081549 | 15,687,562,500.0 |
| **550** | 50 | 3.056677 | 22,959,825,625.0 | 0.100339 | 22,959,825,625.0 |
| **600** | 60 | 4.072283 | 32,508,090,000.0 | 0.149209 | 32,508,090,000.0 |
| **1000** | 100 | 22.626312 | 250,500,250,000.0 | 0.625287 | 250,500,250,000.0 |
| **700** | 70 | 7.611243 | 60,196,622,500.0 | 0.220870 | 60,196,622,500.0 |
| **1000** | 100 | 22.560374 | 250,500,250,000.0 | 0.630177 | 250,500,250,000.0 |
| **800** | 80 | 10.029581 | 102,656,160,000.0 | 0.327374 | 102,656,160,000.0 |
| **1000** | 100 | 22.596446 | 250,500,250,000.0 | 0.638827 | 250,500,250,000.0 |
| **900** | 90 | 20.174155 | 164,389,702,500.0 | 0.450073 | 164,389,702,500.0 |
| **1000** | 100 | 22.571971 | 250,500,250,000.0 | 0.635937 | 250,500,250,000.0 |
| **1000** | 100 | 22.702559 | 250,500,250,000.0 | 0.647013 | 250,500,250,000.0 |

**Graph**

*data values sorted*




## *CONCLUSIONS*

Manual straight-forward IJK computation seems to be consistent in time taken using one or multiple threads when calculated on csserver rather on yeats respectively, even though single processors were used for both. There was a slight increase when using yeats but that could be due to the time of day rather than the actual system. Savings from computing in parallel usually lead to an overall improvement in computation over manual computations. There is, however, an added overhead when using threads of creating and maintaining them as well as switching between them. This was also noticeable, as pthreads computations took less than one second regardless of the size of the matrix and number of threads applicable.

It is expected that the time saving (time is what is being measured here and not number of computations) when using threads to run computations in parallel needs to be greater than the overhead of creating, maintaining, and switching between the threads. Perhaps it would be interesting to evaluate the time taken for exceptionally large matrices of greater than 100,000 x 100,000. However, that was something deemed risky for the systems involved when it took about 20 to 25 seconds to calculate |C| manually.

Dgemm (cblas only) was computed on csserver only in order to complete the results obtained for large and smaller matrix sizes with the time taken for straight-forward IJK computations. It seemed to remain reasonably consistent for smaller matrices of time taken for dgemm : manual of circa 1 : 4.

Using threads when performing read/write operations may cause the the program to wait for a resource, e.g.: file read/write which may slow down processing, this was not noticeable in the computations performed.

Overall, pThreads was faster than a manual straight-forward IJK computation, which was also slower than dgemm to perform the same or similar computations. This was the case for both random and incremental cell value initialization for |A| and |B|. The time taken for manual remained somewhat consistent across cserver and yeats and so could be used as a could comparison for dgemm and pthreads.

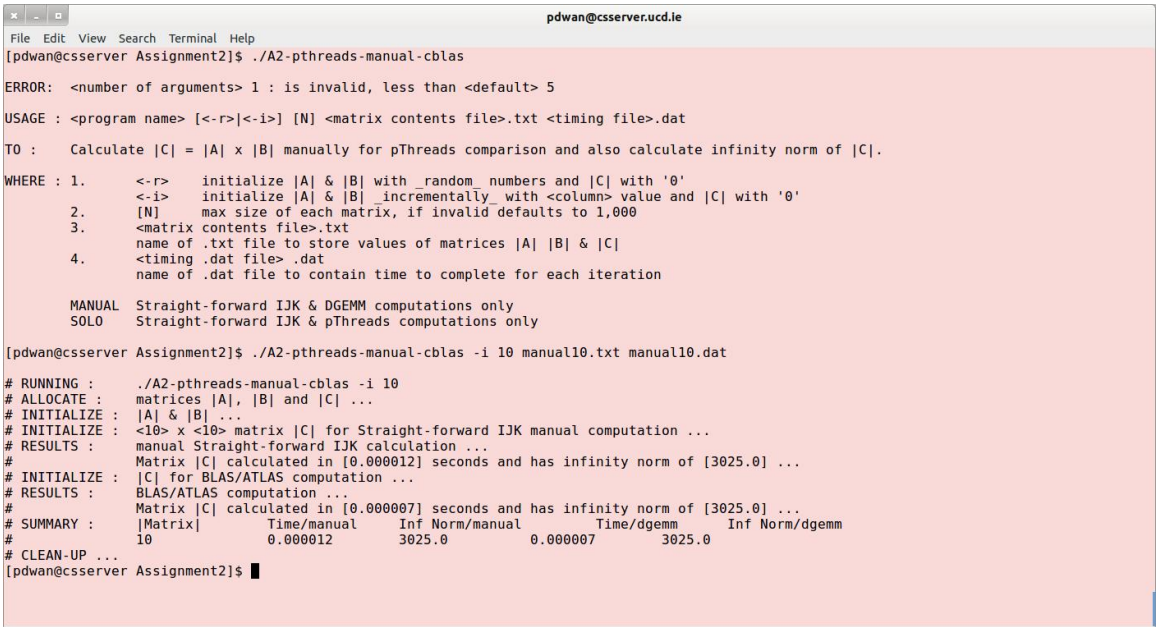
# APPENDICES

## *APPENDIX I – VALIDATE RESULTS*

Spot check only using 10x10 matrices, initializing matrices |A| and |B| using successive column values.

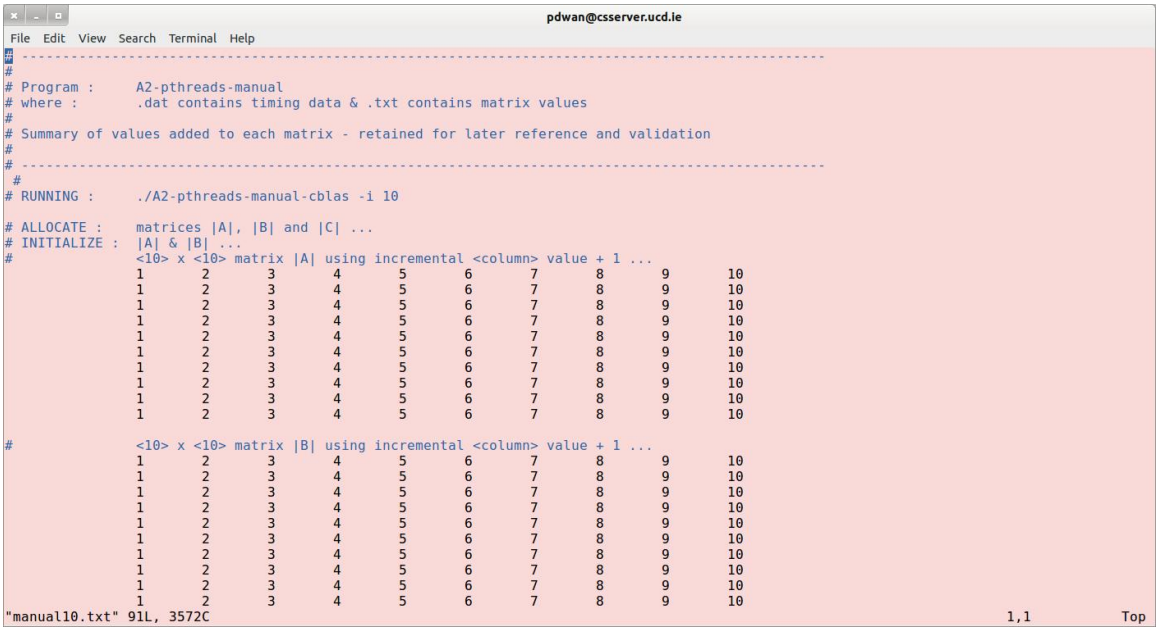
### *A2-pthreads-manual-cblas*

Executed using :

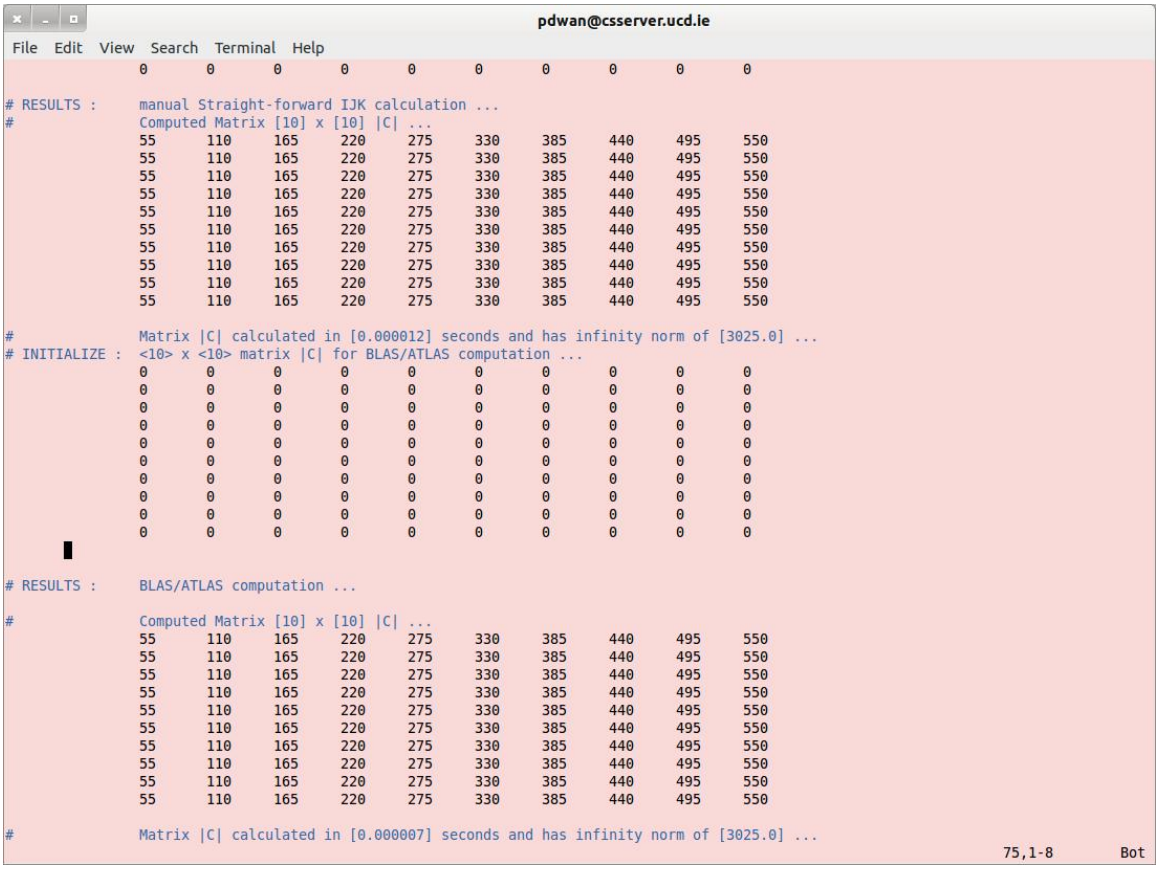


```
pdwan@csserver.ucd.ie
File  Edit  View  Search  Terminal  Help
[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas

ERROR:  <number of arguments> 1 : is invalid, less than <default> 5

USAGE : <program name> [<-r>|<-i>] [N] <matrix contents file>.txt <timing file>.dat

TO :    Calculate |C| = |A| x |B| manually for pThreads comparison and also calculate infinity norm of |C|.

WHERE : 1.      <-r>    initialize |A| & |B| with _random_ numbers and |C| with '0'
                <-i>    initialize |A| & |B| _incrementally_ with <column> value and |C| with '0'
        2.      [N]     max size of each matrix, if invalid defaults to 1,000
        3.      <matrix contents file>.txt
                name of .txt file to store values of matrices |A| |B| & |C|
        4.      <timing .dat file> .dat
                name of .dat file to contain time to complete for each iteration

        MANUAL  Straight-forward IJK & DGEMM computations only
        SOLO    Straight-forward IJK & pThreads computations only

[pdwan@csserver Assignment2]$ ./A2-pthreads-manual-cblas -i 10 manual10.txt manual10.dat

# RUNNING :     ./A2-pthreads-manual-cblas -i 10
# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
# INITIALIZE :  <10> x <10> matrix |C| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |C| calculated in [0.000012] seconds and has infinity norm of [3025.0] ...
# INITIALIZE :  |C| for BLAS/ATLAS computation ...
# RESULTS :     BLAS/ATLAS computation ...
#               Matrix |C| calculated in [0.000007] seconds and has infinity norm of [3025.0] ...
# SUMMARY :     |Matrix|        Time/manual     Inf Norm/manual         Time/dgemm      Inf Norm/dgemm
#               10              0.000012        3025.0          0.000007        3025.0
# CLEAN-UP ...
[pdwan@csserver Assignment2]$
```

Resulting sample Matrix .txt file contains :

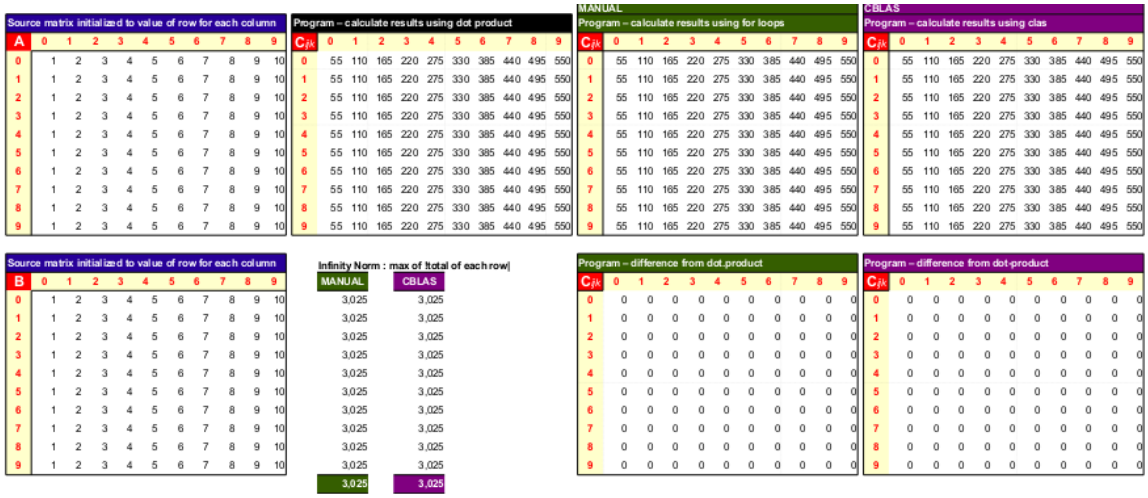


```
pdwan@csserver.ucd.ie
File  Edit  View  Search  Terminal  Help
# ------------------------------------------------------------------------------------------------
#
# Program :     A2-pthreads-manual
# where :       .dat contains timing data & .txt contains matrix values
#
# Summary of values added to each matrix - retained for later reference and validation
#
# ------------------------------------------------------------------------------------------------
 #
# RUNNING :     ./A2-pthreads-manual-cblas -i 10

# ALLOCATE :    matrices |A|, |B| and |C| ...
# INITIALIZE :  |A| & |B| ...
#               <10> x <10> matrix |A| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10

#               <10> x <10> matrix |B| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
"manual10.txt" 91L, 3572C                                                     1,1           Top
```



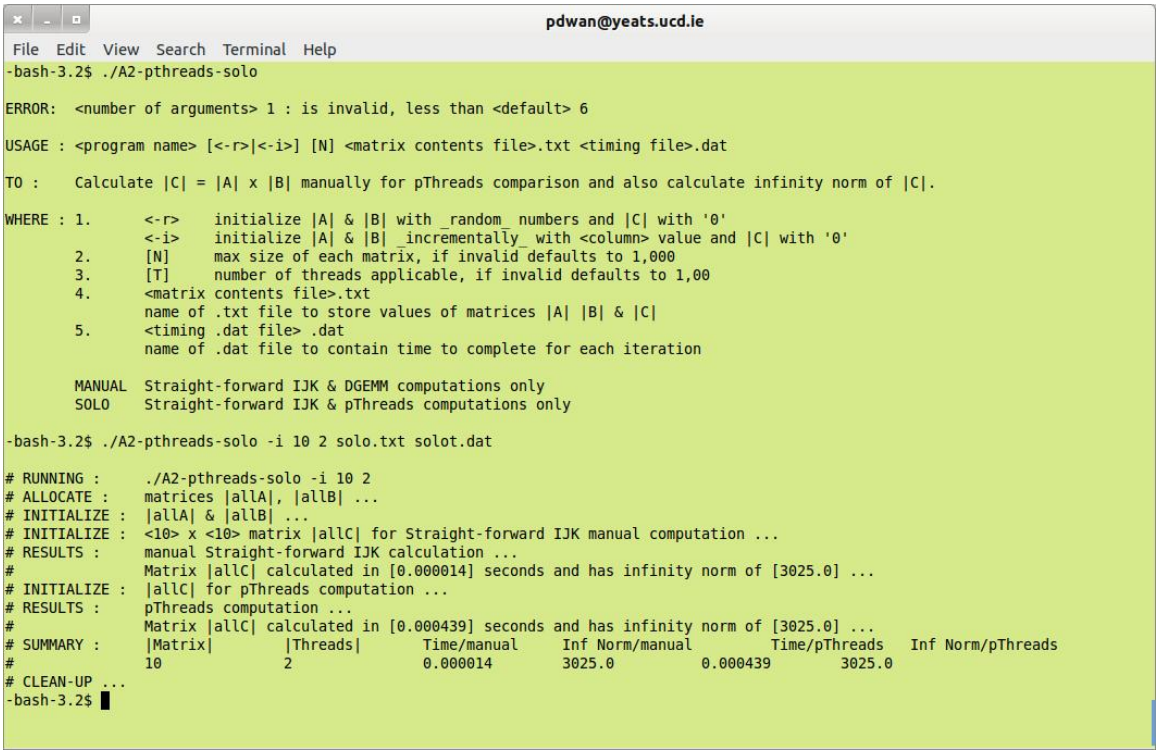

Resulting sample summary timing data file contains :


Validating results for manual computation gives :



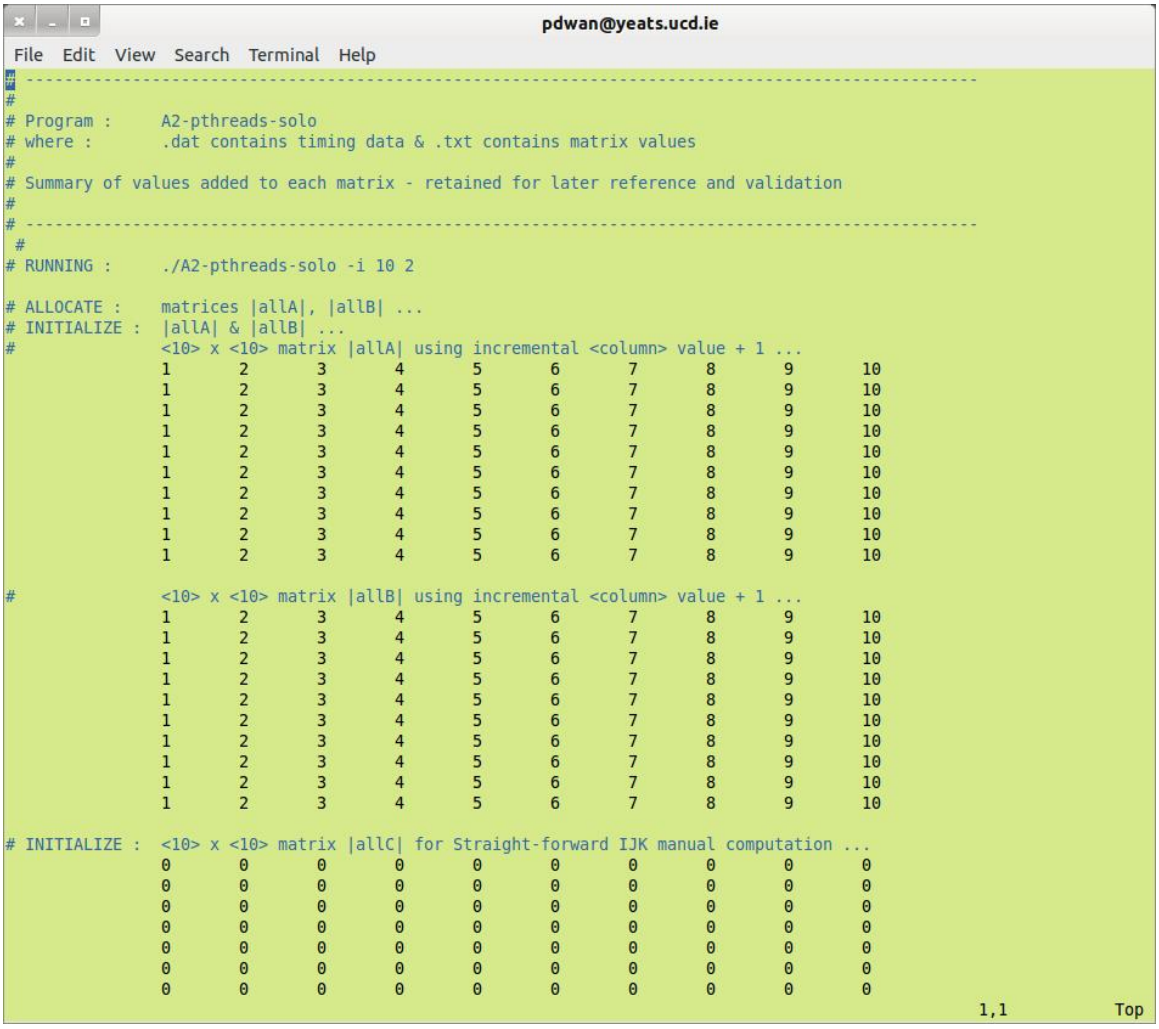
### *A2-pthreads-solo*

Executed using :

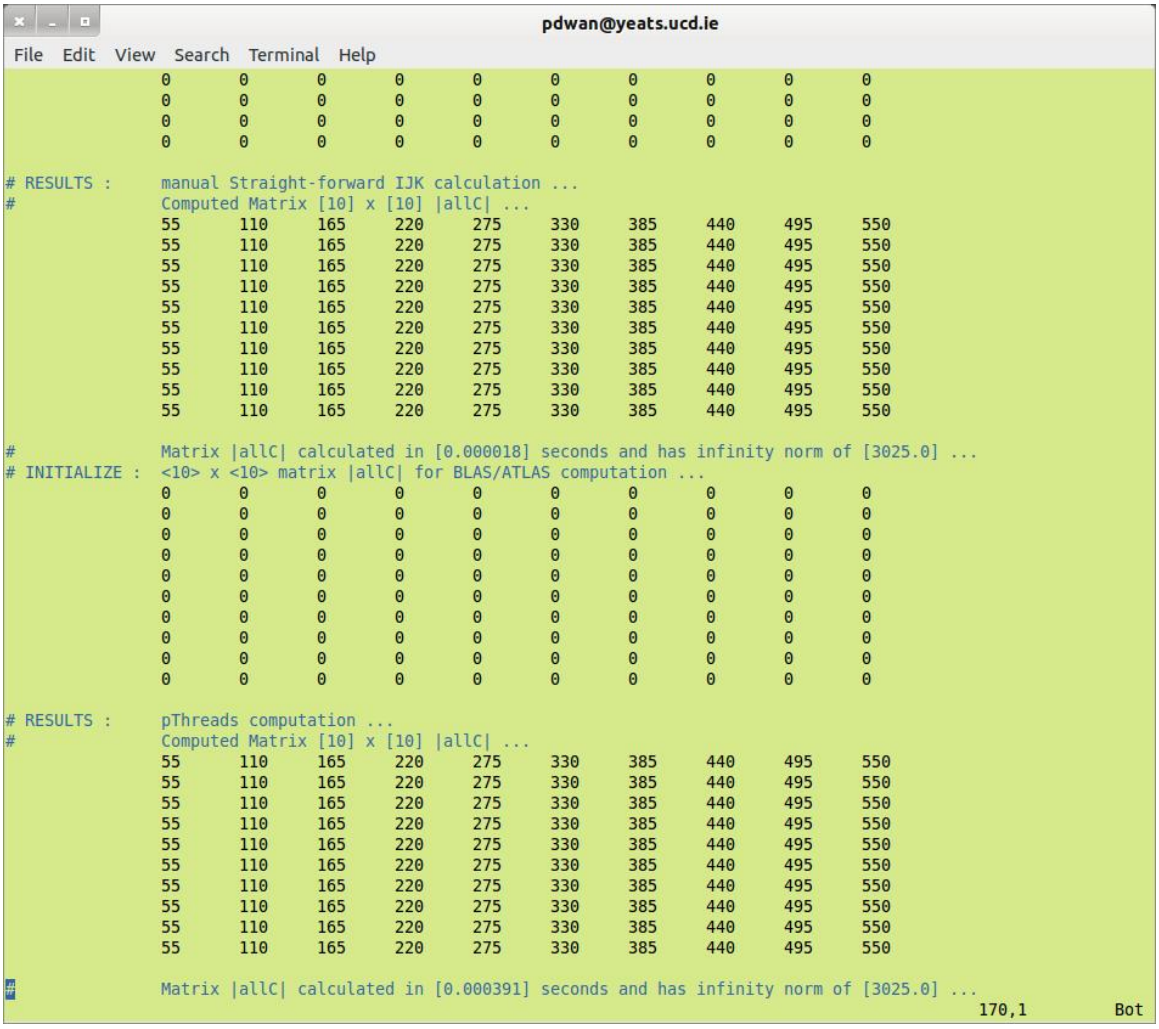


```
pdwan@yeats.ucd.ie
File  Edit  View  Search  Terminal  Help
-bash-3.2$ ./A2-pthreads-solo

ERROR:  <number of arguments> 1 : is invalid, less than <default> 6

USAGE : <program name> [<-r>|<-i>] [N] <matrix contents file>.txt <timing file>.dat

TO :    Calculate |C| = |A| x |B| manually for pThreads comparison and also calculate infinity norm of |C|.

WHERE : 1.      <-r>    initialize |A| & |B| with _random_ numbers and |C| with '0'
                <-i>    initialize |A| & |B| _incrementally_ with <column> value and |C| with '0'
        2.      [N]     max size of each matrix, if invalid defaults to 1,000
        3.      [T]     number of threads applicable, if invalid defaults to 1,00
        4.      <matrix contents file>.txt
                name of .txt file to store values of matrices |A| |B| & |C|
        5.      <timing .dat file> .dat
                name of .dat file to contain time to complete for each iteration

        MANUAL  Straight-forward IJK & DGEMM computations only
        SOLO    Straight-forward IJK & pThreads computations only

-bash-3.2$ ./A2-pthreads-solo -i 10 2 solo.txt solot.dat

# RUNNING :     ./A2-pthreads-solo -i 10 2
# ALLOCATE :    matrices |allA|, |allB| ...
# INITIALIZE :  |allA| & |allB| ...
# INITIALIZE :  <10> x <10> matrix |allC| for Straight-forward IJK manual computation ...
# RESULTS :     manual Straight-forward IJK calculation ...
#               Matrix |allC| calculated in [0.000014] seconds and has infinity norm of [3025.0] ...
# INITIALIZE :  |allC| for pThreads computation ...
# RESULTS :     pThreads computation ...
#               Matrix |allC| calculated in [0.000439] seconds and has infinity norm of [3025.0] ...
# SUMMARY :     |Matrix|        |Threads|       Time/manual     Inf Norm/manual         Time/pThreads   Inf Norm/pThreads
#               10              2               0.000014        3025.0          0.000439        3025.0
# CLEAN-UP ...
-bash-3.2$
```

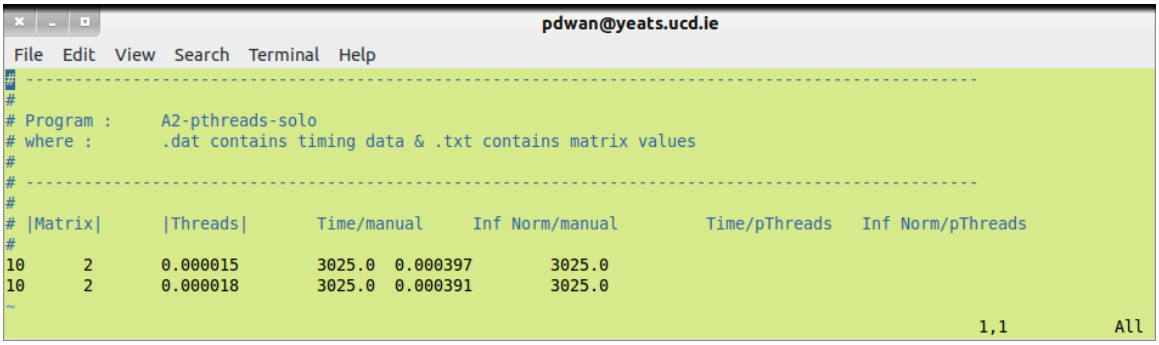
Resulting matrix file contains :



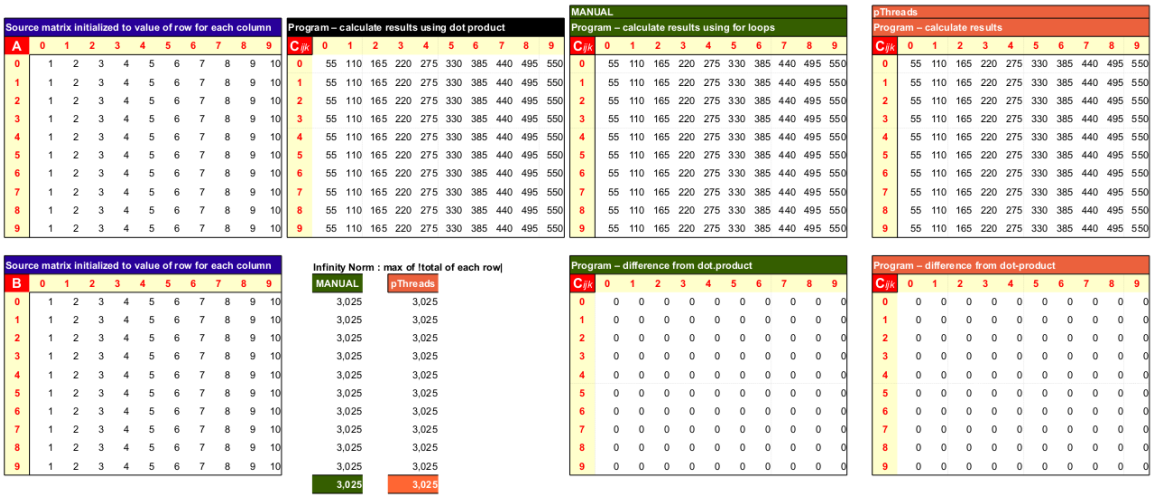
```
pdwan@yeats.ucd.ie
File  Edit  View  Search  Terminal  Help
# ------------------------------------------------------------------------------------------------
#
# Program :     A2-pthreads-solo
# where :       .dat contains timing data & .txt contains matrix values
#
# Summary of values added to each matrix - retained for later reference and validation
#
# ------------------------------------------------------------------------------------------------
 #
# RUNNING :     ./A2-pthreads-solo -i 10 2

# ALLOCATE :    matrices |allA|, |allB| ...
# INITIALIZE :  |allA| & |allB| ...
#               <10> x <10> matrix |allA| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10

#               <10> x <10> matrix |allB| using incremental <column> value + 1 ...
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10
                1       2       3       4       5       6       7       8       9       10

# INITIALIZE :  <10> x <10> matrix |allC| for Straight-forward IJK manual computation ...
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                0       0       0       0       0       0       0       0       0       0
                                                                                        1,1           Top
```




Resulting sample summary timing data file contains *(two runs here)*:




Validating results for pthreads computation gives :



## *APPENDIX II – SUMMARY OF RESULTS OBTAINED*

### *./A2-pthreads-manual-cblas -r 50 → 100 : range 1*

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001202 | 89,502.0 | 0.000275 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001201 | 89,502.0 | 0.000273 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001197 | 89,502.0 | 0.000279 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001201 | 89,502.0 | 0.000277 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001198 | 89,502.0 | 0.000283 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001199 | 89,502.0 | 0.000276 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009303 | 353,266.0 | 0.001979 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009351 | 353,266.0 | 0.001987 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009313 | 353,266.0 | 0.001989 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009438 | 353,266.0 | 0.001982 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009326 | 353,266.0 | 0.001982 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009328 | 353,266.0 | 0.001984 | 353,266.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001195 | 89,502.0 | 0.000277 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001196 | 89,502.0 | 0.000283 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001206 | 89,502.0 | 0.000277 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001238 | 89,502.0 | 0.000280 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001203 | 89,502.0 | 0.000278 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001192 | 89,502.0 | 0.000280 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009321 | 353,266.0 | 0.005735 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009326 | 353,266.0 | 0.002006 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009396 | 353,266.0 | 0.001986 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009311 | 353,266.0 | 0.004446 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009299 | 353,266.0 | 0.001986 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009279 | 353,266.0 | 0.001978 | 353,266.0 |

### *./A2-pthreads-manual-cblas -i 50 → 100 : range 1*

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001259 | 1,625,625.0 | 0.000283 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.002391 | 1,625,625.0 | 0.000295 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001297 | 1,625,625.0 | 0.000287 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001232 | 1,625,625.0 | 0.000280 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001216 | 1,625,625.0 | 0.000291 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001267 | 1,625,625.0 | 0.000289 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.010098 | 25,502,500.0 | 0.002294 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.010646 | 25,502,500.0 | 0.002014 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009306 | 25,502,500.0 | 0.004446 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009266 | 25,502,500.0 | 0.001982 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009307 | 25,502,500.0 | 0.001979 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009287 | 25,502,500.0 | 0.001974 | 25,502,500.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001194 | 1,625,625.0 | 0.000273 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001662 | 1,625,625.0 | 0.000274 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001197 | 1,625,625.0 | 0.000284 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001198 | 1,625,625.0 | 0.000274 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001202 | 1,625,625.0 | 0.000273 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001203 | 1,625,625.0 | 0.000273 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.015590 | 25,502,500.0 | 0.002003 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009303 | 25,502,500.0 | 0.001977 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009298 | 25,502,500.0 | 0.001981 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009405 | 25,502,500.0 | 0.001979 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009290 | 25,502,500.0 | 0.001989 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009297 | 25,502,500.0 | 0.001990 | 25,502,500.0 |

### *./A2-pthreads-manual-cblas -r 50 → 1000 : range 2*

```
#  Matrix & pThreads - range 2
declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001199 | 89,502.0 | 0.000275 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001742 | 89,502.0 | 0.000279 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001195 | 89,502.0 | 0.000278 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009499 | 353,266.0 | 0.001992 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009315 | 353,266.0 | 0.002046 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009298 | 353,266.0 | 0.002074 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.247450 | 8,117,937.0 | 0.653315 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.119148 | 8,117,937.0 | 0.651457 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.311281 | 8,117,937.0 | 0.651434 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.091983 | 8,117,937.0 | 0.652713 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.251307 | 31,896,067.0 | 5.214984 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.300947 | 31,896,067.0 | 5.213675 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.192011 | 31,896,067.0 | 5.229068 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.505856 | 31,896,067.0 | 5.217558 | 31,896,067.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001194 | 89,502.0 | 0.000273 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001195 | 89,502.0 | 0.000274 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001205 | 89,502.0 | 0.000275 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009327 | 353,266.0 | 0.001991 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009324 | 353,266.0 | 0.001989 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 100 | 0.009338 | 353,266.0 | 0.002017 | 353,266.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.055617 | 8,117,937.0 | 0.647558 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.079544 | 8,117,937.0 | 0.647640 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.075784 | 8,117,937.0 | 0.647560 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.079752 | 8,117,937.0 | 0.646994 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.499682 | 31,896,067.0 | 5.126204 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 19.939184 | 31,896,067.0 | 5.101704 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.639626 | 31,896,067.0 | 5.305914 | 31,896,067.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.272806 | 31,896,067.0 | 5.207787 | 31,896,067.0 |

### *./A2-pthreads-manual-cblas -i 50 → 1000 : range 2*

```
#  Matrix & pThreads - range 2
declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001194 | 1,625,625.0 | 0.000275 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001195 | 1,625,625.0 | 0.000277 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001192 | 1,625,625.0 | 0.000279 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009297 | 25,502,500.0 | 0.001984 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009348 | 25,502,500.0 | 0.001983 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009267 | 25,502,500.0 | 0.001976 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.080425 | 15,687,562,500.0 | 0.641401 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.112705 | 15,687,562,500.0 | 0.641102 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.082249 | 15,687,562,500.0 | 0.640499 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.081559 | 15,687,562,500.0 | 0.644471 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.073517 | 250,500,250,000.0 | 5.152608 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.013564 | 250,500,250,000.0 | 5.123747 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.607161 | 250,500,250,000.0 | 5.152919 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.096499 | 250,500,250,000.0 | 5.164086 | 250,500,250,000.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001222 | 1,625,625.0 | 0.000274 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001200 | 1,625,625.0 | 0.000274 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001208 | 1,625,625.0 | 0.000278 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009360 | 25,502,500.0 | 0.001993 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009315 | 25,502,500.0 | 0.001991 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 100 | 0.009289 | 25,502,500.0 | 0.001983 | 25,502,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.041600 | 15,687,562,500.0 | 0.645565 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.163960 | 15,687,562,500.0 | 0.644723 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.078707 | 15,687,562,500.0 | 0.640226 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.076091 | 15,687,562,500.0 | 0.643444 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.266145 | 250,500,250,000.0 | 5.121193 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.055608 | 250,500,250,000.0 | 5.103926 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 19.965258 | 250,500,250,000.0 | 5.144229 | 250,500,250,000.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 19.876712 | 250,500,250,000.0 | 5.094854 | 250,500,250,000.0 |

### *./A2-pthreads-manual-cblas -r 50 → 1000 : range 3*

```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001299 | 89,502.0 | 0.000307 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 150 | 0.037241 | 772,112.0 | 0.007939 | 772,112.0 |
| *./A2-pthreads-manual-cblas -r* | 200 | 0.124785 | 1,322,244.0 | 0.025349 | 1,322,244.0 |
| *./A2-pthreads-manual-cblas -r* | 250 | 0.341618 | 2,081,836.0 | 0.064024 | 2,081,836.0 |
| *./A2-pthreads-manual-cblas -r* | 300 | 0.612026 | 3,011,393.0 | 0.132935 | 3,011,393.0 |
| *./A2-pthreads-manual-cblas -r* | 350 | 0.986115 | 4,095,433.0 | 0.222378 | 4,095,433.0 |
| *./A2-pthreads-manual-cblas -r* | 400 | 1.081713 | 5,225,598.0 | 0.330024 | 5,225,598.0 |
| *./A2-pthreads-manual-cblas -r* | 450 | 2.224804 | 6,598,115.0 | 0.469677 | 6,598,115.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.155695 | 8,117,937.0 | 0.646205 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 550 | 4.036771 | 9,746,097.0 | 0.856691 | 9,746,097.0 |
| *./A2-pthreads-manual-cblas -r* | 600 | 4.280408 | 11,584,750.0 | 1.107333 | 11,584,750.0 |
| *./A2-pthreads-manual-cblas -r* | 650 | 6.654927 | 13,574,185.0 | 1.407244 | 13,574,185.0 |
| *./A2-pthreads-manual-cblas -r* | 700 | 8.415577 | 15,701,890.0 | 1.757862 | 15,701,890.0 |
| *./A2-pthreads-manual-cblas -r* | 750 | 10.446816 | 18,100,461.0 | 2.172926 | 18,100,461.0 |
| *./A2-pthreads-manual-cblas -r* | 800 | 10.345640 | 20,461,090.0 | 2.631782 | 20,461,090.0 |
| *./A2-pthreads-manual-cblas -r* | 850 | 14.921297 | 23,321,446.0 | 3.154325 | 23,321,446.0 |
| *./A2-pthreads-manual-cblas -r* | 900 | 18.114296 | 25,890,261.0 | 3.742948 | 25,890,261.0 |
| *./A2-pthreads-manual-cblas -r* | 950 | 20.963989 | 28,863,806.0 | 4.391238 | 28,863,806.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.036045 | 31,896,067.0 | 5.105140 | 31,896,067.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -r* | 50 | 0.001198 | 89,502.0 | 0.000277 | 89,502.0 |
| *./A2-pthreads-manual-cblas -r* | 150 | 0.031242 | 772,112.0 | 0.006532 | 772,112.0 |
| *./A2-pthreads-manual-cblas -r* | 200 | 0.094799 | 1,322,244.0 | 0.016427 | 1,322,244.0 |
| *./A2-pthreads-manual-cblas -r* | 250 | 0.285936 | 2,081,836.0 | 0.059638 | 2,081,836.0 |
| *./A2-pthreads-manual-cblas -r* | 300 | 0.597465 | 3,011,393.0 | 0.133201 | 3,011,393.0 |
| *./A2-pthreads-manual-cblas -r* | 350 | 0.996649 | 4,095,433.0 | 0.223727 | 4,095,433.0 |
| *./A2-pthreads-manual-cblas -r* | 400 | 1.081898 | 5,225,598.0 | 0.331603 | 5,225,598.0 |
| *./A2-pthreads-manual-cblas -r* | 450 | 2.195723 | 6,598,115.0 | 0.474050 | 6,598,115.0 |
| *./A2-pthreads-manual-cblas -r* | 500 | 3.051341 | 8,117,937.0 | 0.647024 | 8,117,937.0 |
| *./A2-pthreads-manual-cblas -r* | 550 | 4.164886 | 9,746,097.0 | 0.862614 | 9,746,097.0 |
| *./A2-pthreads-manual-cblas -r* | 600 | 4.426588 | 11,584,750.0 | 1.114165 | 11,584,750.0 |
| *./A2-pthreads-manual-cblas -r* | 650 | 6.704661 | 13,574,185.0 | 1.412377 | 13,574,185.0 |
| *./A2-pthreads-manual-cblas -r* | 700 | 8.479324 | 15,701,890.0 | 1.761060 | 15,701,890.0 |
| *./A2-pthreads-manual-cblas -r* | 750 | 10.389538 | 18,100,461.0 | 2.169991 | 18,100,461.0 |
| *./A2-pthreads-manual-cblas -r* | 800 | 10.622015 | 20,461,090.0 | 2.646453 | 20,461,090.0 |
| *./A2-pthreads-manual-cblas -r* | 850 | 15.084184 | 23,321,446.0 | 3.156021 | 23,321,446.0 |
| *./A2-pthreads-manual-cblas -r* | 900 | 18.183891 | 25,890,261.0 | 3.751744 | 25,890,261.0 |
| *./A2-pthreads-manual-cblas -r* | 950 | 21.633958 | 28,863,806.0 | 4.388081 | 28,863,806.0 |
| *./A2-pthreads-manual-cblas -r* | 1000 | 20.132292 | 31,896,067.0 | 5.122467 | 31,896,067.0 |

### *./A2-pthreads-manual-cblas -i 50 → 1000 : range 3*

```
#  Matrix & pThreads - range 3
   declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
   declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

| *Using* | \|Matrix\| | Time / manual | Inf Norm / manual | Time / dgemm | Inf Norm / dgemm |
|---|---|---|---|---|---|
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001193 | 1,625,625.0 | 0.000272 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 150 | 0.062508 | 128,255,625.0 | 0.006597 | 128,255,625.0 |
| *./A2-pthreads-manual-cblas -i* | 200 | 0.093369 | 404,010,000.0 | 0.016680 | 404,010,000.0 |
| *./A2-pthreads-manual-cblas -i* | 250 | 0.352370 | 984,390,625.0 | 0.072505 | 984,390,625.0 |
| *./A2-pthreads-manual-cblas -i* | 300 | 0.652539 | 2,038,522,500.0 | 0.126024 | 2,038,522,500.0 |
| *./A2-pthreads-manual-cblas -i* | 350 | 0.993798 | 3,773,030,625.0 | 0.218930 | 3,773,030,625.0 |
| *./A2-pthreads-manual-cblas -i* | 400 | 1.081100 | 6,432,040,000.0 | 0.330587 | 6,432,040,000.0 |
| *./A2-pthreads-manual-cblas -i* | 450 | 2.387546 | 10,297,175,625.0 | 0.476119 | 10,297,175,625.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.082533 | 15,687,562,500.0 | 0.644611 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 550 | 4.249742 | 22,959,825,625.0 | 0.858860 | 22,959,825,625.0 |
| *./A2-pthreads-manual-cblas -i* | 600 | 4.298823 | 32,508,090,000.0 | 1.112652 | 32,508,090,000.0 |
| *./A2-pthreads-manual-cblas -i* | 650 | 6.642814 | 44,763,980,625.0 | 1.421403 | 44,763,980,625.0 |
| *./A2-pthreads-manual-cblas -i* | 700 | 8.440308 | 60,196,622,500.0 | 1.762349 | 60,196,622,500.0 |
| *./A2-pthreads-manual-cblas -i* | 750 | 10.307231 | 79,312,640,625.0 | 2.180169 | 79,312,640,625.0 |
| *./A2-pthreads-manual-cblas -i* | 800 | 9.210082 | 102,656,160,000.0 | 2.662343 | 102,656,160,000.0 |
| *./A2-pthreads-manual-cblas -i* | 850 | 15.091394 | 130,808,805,625.0 | 3.157122 | 130,808,805,625.0 |
| *./A2-pthreads-manual-cblas -i* | 900 | 18.004254 | 164,389,702,500.0 | 3.754617 | 164,389,702,500.0 |
| *./A2-pthreads-manual-cblas -i* | 950 | 21.020247 | 204,055,475,625.0 | 4.394798 | 204,055,475,625.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.269382 | 250,500,250,000.0 | 5.117270 | 250,500,250,000.0 |
| | | | | | |
| *./A2-pthreads-manual-cblas -i* | 50 | 0.001194 | 1,625,625.0 | 0.000273 | 1,625,625.0 |
| *./A2-pthreads-manual-cblas -i* | 150 | 0.031261 | 128,255,625.0 | 0.006626 | 128,255,625.0 |
| *./A2-pthreads-manual-cblas -i* | 200 | 0.110694 | 404,010,000.0 | 0.015637 | 404,010,000.0 |
| *./A2-pthreads-manual-cblas -i* | 250 | 0.293485 | 984,390,625.0 | 0.060926 | 984,390,625.0 |
| *./A2-pthreads-manual-cblas -i* | 300 | 0.651598 | 2,038,522,500.0 | 0.147015 | 2,038,522,500.0 |
| *./A2-pthreads-manual-cblas -i* | 350 | 1.101349 | 3,773,030,625.0 | 0.250406 | 3,773,030,625.0 |
| *./A2-pthreads-manual-cblas -i* | 400 | 1.085703 | 6,432,040,000.0 | 0.413507 | 6,432,040,000.0 |
| *./A2-pthreads-manual-cblas -i* | 450 | 2.180051 | 10,297,175,625.0 | 0.470420 | 10,297,175,625.0 |
| *./A2-pthreads-manual-cblas -i* | 500 | 3.079099 | 15,687,562,500.0 | 0.643537 | 15,687,562,500.0 |
| *./A2-pthreads-manual-cblas -i* | 550 | 4.220882 | 22,959,825,625.0 | 0.852399 | 22,959,825,625.0 |
| *./A2-pthreads-manual-cblas -i* | 600 | 4.286083 | 32,508,090,000.0 | 1.106982 | 32,508,090,000.0 |
| *./A2-pthreads-manual-cblas -i* | 650 | 6.674248 | 44,763,980,625.0 | 1.412180 | 44,763,980,625.0 |
| *./A2-pthreads-manual-cblas -i* | 700 | 8.440920 | 60,196,622,500.0 | 1.765974 | 60,196,622,500.0 |
| *./A2-pthreads-manual-cblas -i* | 750 | 10.329453 | 79,312,640,625.0 | 2.173019 | 79,312,640,625.0 |
| *./A2-pthreads-manual-cblas -i* | 800 | 9.612526 | 102,656,160,000.0 | 2.622674 | 102,656,160,000.0 |
| *./A2-pthreads-manual-cblas -i* | 850 | 15.167497 | 130,808,805,625.0 | 3.142448 | 130,808,805,625.0 |
| *./A2-pthreads-manual-cblas -i* | 900 | 17.957808 | 164,389,702,500.0 | 3.728599 | 164,389,702,500.0 |
| *./A2-pthreads-manual-cblas -i* | 950 | 20.988347 | 204,055,475,625.0 | 4.394569 | 204,055,475,625.0 |
| *./A2-pthreads-manual-cblas -i* | 1000 | 20.032625 | 250,500,250,000.0 | 5.418355 | 250,500,250,000.0 |

### *./A2-pthreads-solo -i 50 → 100 : range 1*

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

| *Using* | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pthreads | Inf Norm / pthreads |
|---|---|---|---|---|---|---|
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001410 | 1,625,625.0 | 0.000794 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001403 | 1,625,625.0 | 0.000768 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001408 | 1,625,625.0 | 0.000615 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001403 | 1,625,625.0 | 0.000956 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001410 | 1,625,625.0 | 0.000710 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001402 | 1,625,625.0 | 0.000883 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011007 | 25,502,500.0 | 0.002675 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011017 | 25,502,500.0 | 0.003503 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011010 | 25,502,500.0 | 0.002168 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011008 | 25,502,500.0 | 0.004088 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011006 | 25,502,500.0 | 0.004361 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011012 | 25,502,500.0 | 0.003443 | 25,502,500.0 |
| | | | | | | |
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001402 | 1,625,625.0 | 0.000780 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001404 | 1,625,625.0 | 0.000777 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 2 | 0.001415 | 1,625,625.0 | 0.000775 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001408 | 1,625,625.0 | 0.000938 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001405 | 1,625,625.0 | 0.000895 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 50 | 5 | 0.001412 | 1,625,625.0 | 0.000936 | 1,625,625.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011072 | 25,502,500.0 | 0.002305 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011012 | 25,502,500.0 | 0.002738 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 10 | 0.011017 | 25,502,500.0 | 0.002255 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011010 | 25,502,500.0 | 0.003405 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011088 | 25,502,500.0 | 0.003177 | 25,502,500.0 |
| *./A2-pthreads-solo -i* | 100 | 20 | 0.011006 | 25,502,500.0 | 0.003435 | 25,502,500.0 |

### *./A2-pthreads-solo -r 50 → 100 : range 1*

```
#  Matrix & pThreads - range 1
   declare -a NXArray=( 50 50 50 50 50 50 100 100 100 100 100 100 )
   declare -a NPArray=( 2 2 2 5 5 5 10 10 10 20 20 20 )
```

| *Using* | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pthreads | Inf Norm / pthreads |
|---|---|---|---|---|---|---|
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001407 | 89,502.0 | 0.000796 | 85,625.0 |
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001412 | 89,502.0 | 0.000797 | 85,625.0 |
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001409 | 89,502.0 | 0.000805 | 85,625.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001417 | 89,502.0 | 0.000868 | 86,323.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001405 | 89,502.0 | 0.000900 | 85,189.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001412 | 89,502.0 | 0.000644 | 85,058.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011013 | 353,266.0 | 0.003046 | 328,456.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011011 | 353,266.0 | 0.003512 | 328,456.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011016 | 353,266.0 | 0.002598 | 328,456.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011008 | 353,266.0 | 0.004229 | 328,456.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011009 | 353,266.0 | 0.004306 | 330,718.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011043 | 353,266.0 | 0.003444 | 322,376.0 |
| | | | | | | |
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001405 | 89,502.0 | 0.000787 | 85,625.0 |
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001409 | 89,502.0 | 0.000783 | 89,502.0 |
| *./A2-pthreads-solo -r* | 50 | 2 | 0.001411 | 89,502.0 | 0.000779 | 85,625.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001406 | 89,502.0 | 0.000940 | 85,189.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001406 | 89,502.0 | 0.000926 | 85,058.0 |
| *./A2-pthreads-solo -r* | 50 | 5 | 0.001405 | 89,502.0 | 0.000918 | 85,058.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011012 | 353,266.0 | 0.003239 | 324,244.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011007 | 353,266.0 | 0.002789 | 328,456.0 |
| *./A2-pthreads-solo -r* | 100 | 10 | 0.011016 | 353,266.0 | 0.002692 | 331,606.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011007 | 353,266.0 | 0.004198 | 330,718.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011006 | 353,266.0 | 0.005866 | 330,718.0 |
| *./A2-pthreads-solo -r* | 100 | 20 | 0.011026 | 353,266.0 | 0.004040 | 326,364.0 |

### *./A2-pthreads-solo -i 50 → 1000 : range 2*

```
# Matrix & pThreads - range 2
declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

| Using | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pthreads | Inf Norm / pthreads |
|---|---|---|---|---|---|---|
| ./A2-pthreads-solo -i | 50 | 10 | 0.001413 | 1,625,625.0 | 0.001308 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001406 | 1,625,625.0 | 0.001294 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001414 | 1,625,625.0 | 0.001214 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011010 | 25,502,500.0 | 0.003006 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011010 | 25,502,500.0 | 0.002655 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011014 | 25,502,500.0 | 0.002412 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.274337 | 15,687,562,500.0 | 0.074011 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.273940 | 15,687,562,500.0 | 0.074282 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.269123 | 15,687,562,500.0 | 0.076757 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.268417 | 15,687,562,500.0 | 0.073420 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.566625 | 250,500,250,000.0 | 0.658890 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.560287 | 250,500,250,000.0 | 0.746149 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.724512 | 250,500,250,000.0 | 0.805147 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.652979 | 250,500,250,000.0 | 0.750033 | 250,500,250,000.0 |
| | | | | | | |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001415 | 1,625,625.0 | 0.001044 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001412 | 1,625,625.0 | 0.001168 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001406 | 1,625,625.0 | 0.001356 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011012 | 25,502,500.0 | 0.003220 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011005 | 25,502,500.0 | 0.002746 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 100 | 10 | 0.011024 | 25,502,500.0 | 0.002686 | 25,502,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.279820 | 15,687,562,500.0 | 0.075656 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.276522 | 15,687,562,500.0 | 0.076434 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.246144 | 15,687,562,500.0 | 0.075013 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 500 | 20 | 2.280677 | 15,687,562,500.0 | 0.078744 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.517391 | 250,500,250,000.0 | 0.751990 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.561444 | 250,500,250,000.0 | 0.941664 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.588215 | 250,500,250,000.0 | 0.736500 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 20 | 22.603145 | 250,500,250,000.0 | 0.723599 | 250,500,250,000.0 |

### *./A2-pthreads-solo -r 50 → 1000 : range 2*

```
# Matrix & pThreads - range 2
declare -a NXArray=( 50 50 50 100 100 100 500 500 500 500 1000 1000 1000 1000 )
declare -a NPArray=( 10 10 10 10 10 10 20 20 20 20 20 20 20 20 )
```

| Using | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pthreads | Inf Norm / pthreads |
|---|---|---|---|---|---|---|
| ./A2-pthreads-solo -r | 50 | 10 | 0.001409 | 89,502.0 | 0.001412 | 83,084.0 |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001408 | 89,502.0 | 0.001331 | 83,084.0 |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001413 | 89,502.0 | 0.001488 | 83,084.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011018 | 353,266.0 | 0.003031 | 324,244.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011017 | 353,266.0 | 0.002967 | 328,456.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011027 | 353,266.0 | 0.003410 | 328,456.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.277439 | 8,117,937.0 | 0.100086 | 7,887,576.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.266145 | 8,117,937.0 | 0.079707 | 7,958,041.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.263969 | 8,117,937.0 | 0.072114 | 7,969,826.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.267955 | 8,117,937.0 | 0.089036 | 7,941,189.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.695424 | 31,896,067.0 | 0.657843 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.644039 | 31,896,067.0 | 0.845433 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.586039 | 31,896,067.0 | 0.761873 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.867197 | 31,896,067.0 | 0.689865 | 31,359,528.0 |
| | | | | | | |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001414 | 89,502.0 | 0.001121 | 83,338.0 |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001416 | 89,502.0 | 0.001249 | 82,995.0 |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001406 | 89,502.0 | 0.001329 | 83,084.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011015 | 353,266.0 | 0.002412 | 328,456.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011013 | 353,266.0 | 0.003042 | 328,456.0 |
| ./A2-pthreads-solo -r | 100 | 10 | 0.011029 | 353,266.0 | 0.003435 | 328,866.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.278701 | 8,117,937.0 | 0.079072 | 7,958,041.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.280150 | 8,117,937.0 | 0.074330 | 7,941,189.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.275920 | 8,117,937.0 | 0.072233 | 7,958,041.0 |
| ./A2-pthreads-solo -r | 500 | 20 | 2.269146 | 8,117,937.0 | 0.073958 | 7,914,706.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.604045 | 31,896,067.0 | 0.805102 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.588872 | 31,896,067.0 | 0.726566 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.591746 | 31,896,067.0 | 0.735695 | 31,359,528.0 |
| ./A2-pthreads-solo -r | 1000 | 20 | 22.595227 | 31,896,067.0 | 0.689487 | 31,359,528.0 |

### *./A2-pthreads-solo -i 50 → 1000 : range 3*

***Note:*** *Some matrix sizes ending in 50 were not always dividable equally by the proposed number of thread. As a result, the defaults of [N] = 1000 and [T] = 100 applied.*

```
#  Matrix & pThreads - range 3
declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

| Using | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pThreads | Inf Norm / pThreads |
|---|---|---|---|---|---|---|
| ./A2-pthreads-solo -i | 50 | 10 | 0.001402 | 1,625,625.0 | 0.001368 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 150 | 10 | 0.036497 | 128,255,625.0 | 0.005516 | 128,255,625.0 |
| ./A2-pthreads-solo -i | 200 | 20 | 0.103509 | 404,010,000.0 | 0.010126 | 404,010,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.578228 | 250,500,250,000.0 | 0.629005 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 300 | 30 | 0.394802 | 2,038,522,500.0 | 0.019049 | 2,038,522,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.531387 | 250,500,250,000.0 | 0.623278 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 400 | 40 | 1.014198 | 6,432,040,000.0 | 0.045393 | 6,432,040,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.699620 | 250,500,250,000.0 | 0.637117 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 500 | 50 | 2.278428 | 15,687,562,500.0 | 0.081549 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 550 | 50 | 3.056677 | 22,959,825,625.0 | 0.100339 | 22,959,825,625.0 |
| ./A2-pthreads-solo -i | 600 | 60 | 4.072283 | 32,508,090,000.0 | 0.149209 | 32,508,090,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.626312 | 250,500,250,000.0 | 0.625287 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 700 | 70 | 7.611243 | 60,196,622,500.0 | 0.220870 | 60,196,622,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.560374 | 250,500,250,000.0 | 0.630177 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 800 | 80 | 10.029581 | 102,656,160,000.0 | 0.327374 | 102,656,160,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.596446 | 250,500,250,000.0 | 0.638827 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 900 | 90 | 20.174155 | 164,389,702,500.0 | 0.450073 | 164,389,702,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.571971 | 250,500,250,000.0 | 0.635937 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.702559 | 250,500,250,000.0 | 0.647013 | 250,500,250,000.0 |
| | | | | | | |
| ./A2-pthreads-solo -i | 50 | 10 | 0.001406 | 1,625,625.0 | 0.001293 | 1,625,625.0 |
| ./A2-pthreads-solo -i | 150 | 10 | 0.036443 | 128,255,625.0 | 0.004162 | 128,255,625.0 |
| ./A2-pthreads-solo -i | 200 | 20 | 0.102373 | 404,010,000.0 | 0.012016 | 404,010,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.590978 | 250,500,250,000.0 | 0.632697 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 300 | 30 | 0.385407 | 2,038,522,500.0 | 0.021244 | 2,038,522,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.636475 | 250,500,250,000.0 | 0.635413 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 400 | 40 | 1.028510 | 6,432,040,000.0 | 0.049324 | 6,432,040,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.667234 | 250,500,250,000.0 | 0.629788 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 500 | 50 | 2.267032 | 15,687,562,500.0 | 0.082868 | 15,687,562,500.0 |
| ./A2-pthreads-solo -i | 550 | 50 | 3.048547 | 22,959,825,625.0 | 0.104482 | 22,959,825,625.0 |
| ./A2-pthreads-solo -i | 600 | 60 | 3.844227 | 32,508,090,000.0 | 0.144682 | 32,508,090,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.577241 | 250,500,250,000.0 | 0.634571 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 700 | 70 | 8.153411 | 60,196,622,500.0 | 0.218322 | 60,196,622,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.671438 | 250,500,250,000.0 | 0.635316 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 800 | 80 | 10.527717 | 102,656,160,000.0 | 0.342593 | 102,656,160,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.648630 | 250,500,250,000.0 | 0.642568 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 900 | 90 | 20.292053 | 164,389,702,500.0 | 0.468552 | 164,389,702,500.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.644521 | 250,500,250,000.0 | 0.639157 | 250,500,250,000.0 |
| ./A2-pthreads-solo -i | 1000 | 100 | 22.637018 | 250,500,250,000.0 | 0.628302 | 250,500,250,000.0 |

### *./A2-pthreads-solo -r 50 → 1000 : range 3*

***Note:*** *Some matrix sizes ending in 50 were not always dividable equally by the proposed number of thread. As a result, the defaults of [N] = 1000 and [T] = 100 applied.*

```
#  Matrix & pThreads - range 3
declare -a NXArray=( 50, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000 )
declare -a NPArray=( 10, 10, 20, 20, 30, 30, 40, 40, 50, 50, 60, 60, 70, 70, 80, 80, 90, 90, 100 )
```

| Using | \|Matrix\| | \|Threads\| | Time / manual | Inf Norm / manual | Time / pThreads | Inf Norm / pThreads |
|---|---|---|---|---|---|---|
| ./A2-pthreads-solo -r | 50 | 10 | 0.001412 | 89,502.0 | 0.001368 | 83,084.0 |
| ./A2-pthreads-solo -r | 150 | 10 | 0.036468 | 772,112.0 | 0.004344 | 718,234.0 |
| ./A2-pthreads-solo -r | 200 | 20 | 0.103250 | 1,322,244.0 | 0.009416 | 1,290,780.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.577512 | 31,896,067.0 | 0.653281 | 31,341,889.0 |
| ./A2-pthreads-solo -r | 300 | 30 | 0.398502 | 3,011,393.0 | 0.020857 | 2,867,749.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.658816 | 31,896,067.0 | 0.668295 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 400 | 40 | 1.030502 | 5,225,598.0 | 0.045610 | 5,105,882.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.634818 | 31,896,067.0 | 0.647604 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 500 | 50 | 2.276678 | 8,117,937.0 | 0.089903 | 7,957,767.0 |
| ./A2-pthreads-solo -r | 550 | 50 | 3.018685 | 9,746,097.0 | 0.108451 | 9,519,842.0 |
| ./A2-pthreads-solo -r | 600 | 60 | 3.970175 | 11,584,750.0 | 0.158750 | 11,512,140.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.625830 | 31,896,067.0 | 0.627810 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 700 | 70 | 7.715003 | 15,701,890.0 | 0.233204 | 15,534,400.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.644341 | 31,896,067.0 | 0.631318 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 800 | 80 | 9.991302 | 20,461,090.0 | 0.322235 | 20,204,923.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 23.727530 | 31,896,067.0 | 0.653829 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 900 | 90 | 20.144580 | 25,890,261.0 | 0.463994 | 25,133,012.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.567932 | 31,896,067.0 | 0.626697 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.602344 | 31,896,067.0 | 0.639306 | 31,350,390.0 |
| | | | | | | |
| ./A2-pthreads-solo -r | 50 | 10 | 0.001402 | 89,502.0 | 0.001367 | 83,084.0 |
| ./A2-pthreads-solo -r | 150 | 10 | 0.036408 | 772,112.0 | 0.004830 | 718,234.0 |
| ./A2-pthreads-solo -r | 200 | 20 | 0.103282 | 1,322,244.0 | 0.008834 | 1,290,780.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.599911 | 31,896,067.0 | 0.618144 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 300 | 30 | 0.390335 | 3,011,393.0 | 0.021323 | 2,867,749.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.589421 | 31,896,067.0 | 0.649053 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 400 | 40 | 1.033686 | 5,225,598.0 | 0.043501 | 5,105,882.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.610600 | 31,896,067.0 | 0.637181 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 500 | 50 | 2.317871 | 8,117,937.0 | 0.086061 | 7,957,767.0 |
| ./A2-pthreads-solo -r | 550 | 50 | 2.994025 | 9,746,097.0 | 0.104394 | 9,519,842.0 |
| ./A2-pthreads-solo -r | 600 | 60 | 3.990792 | 11,584,750.0 | 0.157564 | 11,512,140.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.655217 | 31,896,067.0 | 0.637070 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 700 | 70 | 7.829027 | 15,701,890.0 | 0.223619 | 15,534,400.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.715330 | 31,896,067.0 | 0.630564 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 800 | 80 | 9.961606 | 20,461,090.0 | 0.323325 | 20,204,923.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.775118 | 31,896,067.0 | 0.632430 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 900 | 90 | 20.011760 | 25,890,261.0 | 0.448777 | 25,133,012.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.545957 | 31,896,067.0 | 0.632317 | 31,350,390.0 |
| ./A2-pthreads-solo -r | 1000 | 100 | 22.624936 | 31,896,067.0 | 0.648349 | 31,350,390.0 |

## *APPENDIX III – REFERENCES / ACKNOWLEDGEMENTS*


- www.stackoverflow.com : general queries on pThreads functionality not working and possible workarounds / solutions
- http://en.wikipedia.org/wiki/Basic_Linear_Algebra_Subprograms
- www.ucd.ie : COMP40700 High Performance Computing – Notes 2014 "pThreads"
- http://blog.speedgocomputing.com/search/label/PThread