Scientific Software Report: Assignment 3

Daniel Loscos Barroso
<daniel.loscosbarroso@student.kuleuven.be>

November 18, 2018

`Time spent:  14 hours`

Discussion of the results

Compiler differences

Each algorithm was measured 3 times for each compiler and N value, this allowed to discard outlaying measurements (which appeared due to other users remotely using the PC I was using at the time.) the 3 measurements are taken one after the other, which may improve the times for the second and third measurements duw to the cache memory status.

The results obtained show that nagfor is the slowest compiler but also the more accurate overall. Gfortran had the best results for the implicit `matmul` function but on the other algorithms ifort was faster. Also, ifort crashed when running the custom method for $N = 1000$ due to a segmentation fault error (probably because of my recursive implementation).

The conclusion is that since `matmul` seems to be the best performing algorithm and it performs better in gfortran; gfortran will be our compiler of reference for the rest of the assignment.

Optimization flags

We used the same testing method then in the previous section.

For $N = 100$ the differences in time between optimization levels where almost non existent for most of the algorithms. However, for the implicit method O1 and O2 performed better than O0. Curiously enough, O4 did not perform better.

The test was repeated with $N = 500$ and similar results were obtained with the different optimization settings. The vector and dot product operations seemed to perform better on O1 and O2 than in O0 and O3. Also O3 showed better results than O0 for the vector operations, but similar or even slightly worse results for the other tests.

My conclusion is that O3 shows the best results overall.

Fastest 3-loop method: JKI

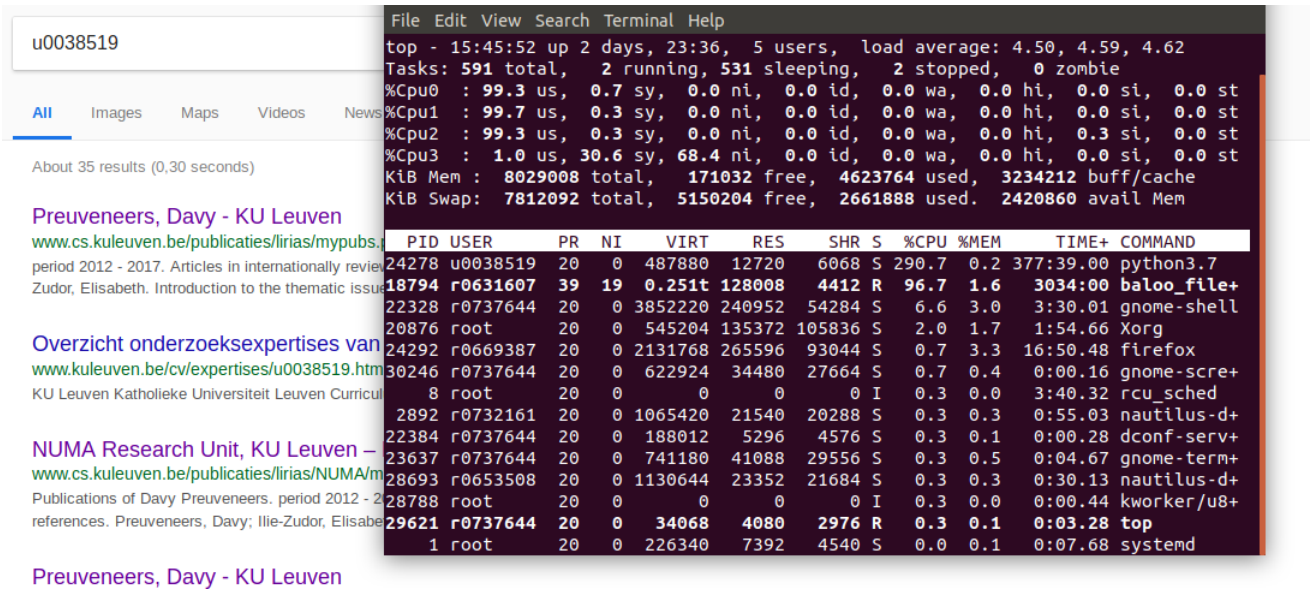
Matrices in fortran are stored in a column-major order. Therefore, to have better memory access time we should try accessing data in a column wise manner (to minimize the number of memory blocks we get in and out of the cache). Makes sense then for i (the only row index) to be on the inner loop (changing more often). Also the outer loop should belong to J (the column index for B) since we only need to access each column for n values of C (better do them all one after the other) while we have to go through all the columns in A for every single value of C.

Fastest dot-product method: Transpose JI

Following the same reasoning as before, I should be in the inner loop. As for why is the transpose faster: we have to go through all the columns in A for every single value of C, but if we transpose it a single column of A transpose contains all the information we need for the dot product, reducing the amount of cache conflicts and improving performance.

Is BLAS using multiple cores in my machine?

Probably it is, but it was impossible to check: I used the `top` command on my terminal while running the process and pressed 1 to check CPU usage in the different cores. Sadly, Mr. Davy Preuveneers was using 3 of my cores during the experiment and I was unable to obtain clear result.



Influence of block size

My L1d cache has 32Kb memory and the L2 cache has 256Kb which is the same size as 2000 and 16000 double precision real numbers respectively. We need to store 4 matrices per block operation (2 operands, 1 partial and 1 final result) so we are looking for $max(x \in \mathbb{N})||4{*}x^2 \leq 2000$ and $max(x \in \mathbb{N})||4{*}x^2 \leq 16000$, these numbers are 22 and 63 respectively.

I wanted to test which of those to numbers worked better and I run a text for $N = 1386$, the result was favorable to the L1d cache number: 22. So this should be our optimal cache size. (More tests with different matrix sizes were made and 22 always outperformed other block sizes, these were not recorded though.)

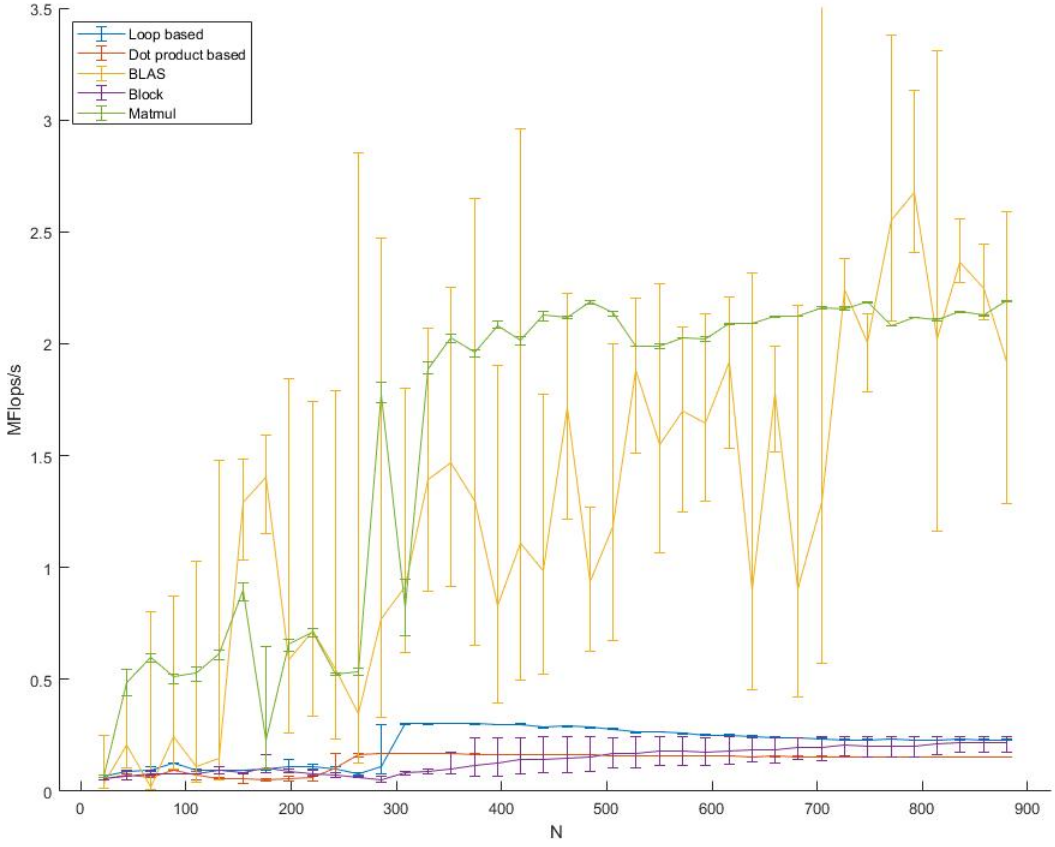
About my own algorithm

Even if it has better asymptotic convergence for low values of N it performs way worse than the regular cubic algorithms. According to Wikipedia, a good fix to my algorithm would be to keep the recursion for block sizes bigger than 3000 and call `matmul` at that point. Te biggest loss of time is made in the recursive call for low values of N.

Comparison of optimal methods

Each product of two $n \times n$ matrices takes $(2n^3 - n^2)$ floating point operations. Even with some abnormalities in the data gathered from the experiments (probably due to remote users), we can notice BLAS and `matmul` clearly outperforming the other methods. The erratic behaviour of BLAS probably has to do with the fact that it uses several CPU cores and three of them where being used by Mr. Preuveneers, the computers in the PC lab are configured to prioritize the local user versus remote users, my theory is that after each measurement my process had claimed more computational power in the other cores and that explains why each measurement is faster than the previous one for the same N.

If my theory holds, we should only look at the higher end of the BLAS error bars and BLAS would outperform every other method.



Raw Results

All the raw results can be found in the folder `results`.

gfortran N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
```

```
IJK: 3.90E-03 sec 3.88E-03 sec 3.90E-03 sec
relative error = 2.98E-16
IKJ: 4.06E-03 sec 4.19E-03 sec 4.10E-03 sec
relative error = 2.98E-16
JIK: 3.99E-03 sec 4.13E-03 sec 3.97E-03 sec
relative error = 2.98E-16
JKI: 3.82E-03 sec 3.83E-03 sec 3.86E-03 sec
relative error = 2.98E-16
KIJ: 3.99E-03 sec 4.01E-03 sec 4.02E-03 sec
relative error = 2.98E-16
KJI: 3.87E-03 sec 3.86E-03 sec 3.85E-03 sec
relative error = 2.98E-16
IKJ, J VECT: 1.66E-03 sec 1.65E-03 sec 1.66E-03 sec
relative error = 2.98E-16
JKI, I VECT: 1.55E-03 sec 1.56E-03 sec 1.55E-03 sec
relative error = 2.98E-16
KIJ, J VECT: 1.61E-03 sec 1.62E-03 sec 1.61E-03 sec
relative error = 2.98E-16
KJI, I VECT: 1.56E-03 sec 1.56E-03 sec 1.55E-03 sec
relative error = 2.98E-16
IJ DOT_PRODUCT: 1.19E-03 sec 1.18E-03 sec 1.18E-03 sec
relative error = 2.98E-16
JI DOT_PRODUCT: 1.21E-03 sec 1.22E-03 sec 1.20E-03 sec
relative error = 2.98E-16
IJ TP DOT_PRODUCT: 1.20E-03 sec 1.18E-03 sec 1.18E-03 sec
relative error = 2.98E-16
JI TP DOT_PRODUCT: 1.19E-03 sec 1.20E-03 sec 1.19E-03 sec
relative error = 2.98E-16
BLAS DGEMM: 6.51E-03 sec 2.23E-04 sec 1.94E-04 sec
relative error = 2.92E-16
IN BLOCKS: 3.78E-03 sec 2.66E-03 sec 4.56E-03 sec
relative error = 1.34E-16
MATMUL: 3.31E-04 sec 2.99E-04 sec 2.98E-04 sec
relative error = 0.00E+00
OWN VARIANT: 9.93E-02 sec 7.60E-02 sec 4.46E-02 sec
relative error = 3.42E-15
```

nagfor N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 3.89E-03 sec 3.89E-03 sec 3.90E-03 sec
relative error = 0.00E+00
IKJ: 4.19E-03 sec 4.03E-03 sec 4.02E-03 sec
relative error = 0.00E+00
JIK: 3.99E-03 sec 4.16E-03 sec 3.96E-03 sec
relative error = 0.00E+00
JKI: 3.79E-03 sec 3.80E-03 sec 3.79E-03 sec
relative error = 0.00E+00
KIJ: 3.96E-03 sec 4.00E-03 sec 4.00E-03 sec
relative error = 0.00E+00
KJI: 3.82E-03 sec 3.82E-03 sec 3.82E-03 sec
relative error = 0.00E+00
IKJ, J VECT: 3.07E-03 sec 3.08E-03 sec 3.11E-03 sec
relative error = 0.00E+00
JKI, I VECT: 2.86E-03 sec 2.88E-03 sec 2.91E-03 sec
relative error = 0.00E+00
KIJ, J VECT: 3.07E-03 sec 3.05E-03 sec 3.08E-03 sec
relative error = 0.00E+00
KJI, I VECT: 2.89E-03 sec 2.90E-03 sec 2.88E-03 sec
relative error = 0.00E+00
IJ DOT_PRODUCT: 2.74E-03 sec 2.74E-03 sec 2.72E-03 sec
relative error = 0.00E+00
JI DOT_PRODUCT: 2.82E-03 sec 2.83E-03 sec 2.80E-03 sec
relative error = 0.00E+00
IJ TP DOT_PRODUCT: 2.70E-03 sec 2.69E-03 sec 2.69E-03 sec
relative error = 0.00E+00
JI TP DOT_PRODUCT: 2.68E-03 sec 2.69E-03 sec 2.71E-03 sec
relative error = 0.00E+00
BLAS DGEMM: 1.87E-02 sec 3.45E-04 sec 2.90E-04 sec
relative error = 1.10E-16
IN BLOCKS: 1.07E-02 sec 2.27E-02 sec 2.06E-02 sec
relative error = 2.95E-16
MATMUL: 1.98E-02 sec 1.98E-02 sec 6.99E-03 sec
relative error = 0.00E+00
```

```
OWN VARIANT: 1.01E-01 sec 3.97E-02 sec 2.52E-02 sec
relative error = 3.52E-15
```

ifort N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 2.79E-04 sec 2.57E-04 sec 2.58E-04 sec
relative error = 0.00E+00
IKJ: 2.52E-04 sec 2.65E-04 sec 2.52E-04 sec
relative error = 0.00E+00
JIK: 2.56E-04 sec 2.64E-04 sec 2.72E-04 sec
relative error = 0.00E+00
JKI: 2.63E-04 sec 2.65E-04 sec 2.57E-04 sec
relative error = 0.00E+00
KIJ: 2.64E-04 sec 2.59E-04 sec 2.61E-04 sec
relative error = 0.00E+00
KJI: 2.55E-04 sec 2.57E-04 sec 2.62E-04 sec
relative error = 0.00E+00
IKJ, J VECT: 4.77E-04 sec 4.75E-04 sec 4.90E-04 sec
relative error = 0.00E+00
JKI, I VECT: 2.57E-04 sec 2.61E-04 sec 2.63E-04 sec
relative error = 0.00E+00
KIJ, J VECT: 4.63E-04 sec 4.78E-04 sec 4.74E-04 sec
relative error = 0.00E+00
KJI, I VECT: 2.56E-04 sec 2.67E-04 sec 2.72E-04 sec
relative error = 0.00E+00
IJ DOT_PRODUCT: 3.71E-04 sec 3.99E-04 sec 3.78E-04 sec
relative error = 2.87E-16
JI DOT_PRODUCT: 2.59E-04 sec 2.59E-04 sec 2.62E-04 sec
relative error = 0.00E+00
IJ TP DOT_PRODUCT: 3.90E-04 sec 3.75E-04 sec 3.85E-04 sec
relative error = 2.73E-16
JI TP DOT_PRODUCT: 3.59E-04 sec 3.83E-04 sec 3.71E-04 sec
relative error = 2.73E-16
BLAS DGEMM: 8.45E-03 sec 2.09E-04 sec 1.21E-04 sec
relative error = 1.08E-16
IN BLOCKS: 2.51E-03 sec 2.07E-03 sec 2.14E-03 sec
```

```
relative error = 2.92E-16
MATMUL: 5.54E-04 sec 5.13E-04 sec 5.16E-04 sec
relative error = 0.00E+00
OWN VARIANT: 1.69E-02 sec 1.77E-02 sec 1.88E-02 sec
relative error = 3.98E-15
```

gfortran N=1000, Block=100

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 4.79E+00 sec 4.79E+00 sec 4.78E+00 sec
relative error = 7.91E-16
IKJ: 9.52E+00 sec 9.87E+00 sec 1.03E+01 sec
relative error = 7.91E-16
JIK: 7.32E+00 sec 7.32E+00 sec 7.31E+00 sec
relative error = 7.91E-16
JKI: 3.72E+00 sec 3.72E+00 sec 3.72E+00 sec
relative error = 7.91E-16
KIJ: 1.00E+01 sec 1.00E+01 sec 1.01E+01 sec
relative error = 7.91E-16
KJI: 3.82E+00 sec 3.81E+00 sec 3.81E+00 sec
relative error = 7.91E-16
IKJ, J VECT: 7.92E+00 sec 7.91E+00 sec 7.91E+00 sec
relative error = 7.91E-16
JKI, I VECT: 1.53E+00 sec 1.53E+00 sec 1.53E+00 sec
relative error = 7.91E-16
KIJ, J VECT: 7.52E+00 sec 7.52E+00 sec 7.52E+00 sec
relative error = 7.91E-16
KJI, I VECT: 1.64E+00 sec 1.64E+00 sec 1.64E+00 sec
relative error = 7.91E-16
IJ DOT_PRODUCT: 2.02E+00 sec 2.03E+00 sec 2.03E+00 sec
relative error = 7.91E-16
JI DOT_PRODUCT: 2.62E+00 sec 2.62E+00 sec 2.62E+00 sec
relative error = 7.91E-16
IJ TP DOT_PRODUCT: 1.27E+00 sec 1.26E+00 sec 1.26E+00 sec
relative error = 7.91E-16
JI TP DOT_PRODUCT: 1.26E+00 sec 1.25E+00 sec 1.25E+00 sec
relative error = 7.91E-16
```

```
BLAS DGEMM: 5.93E-02 sec 6.56E-02 sec 6.92E-02 sec
relative error = 2.34E-16
IN BLOCKS: 1.37E+00 sec 1.23E+00 sec 1.23E+00 sec
relative error = 1.50E-16
MATMUL: 8.29E-02 sec 8.36E-02 sec 8.46E-02 sec
relative error = 0.00E+00
OWN VARIANT: 5.08E+00 sec 5.09E+00 sec 5.09E+00 sec
relative error = 2.20E-14
```

nagfor N=1000, Block=100

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 5.27E+00 sec 5.31E+00 sec 5.23E+00 sec
relative error = 0.00E+00
IKJ: 1.05E+01 sec 1.04E+01 sec 1.04E+01 sec
relative error = 0.00E+00
JIK: 7.39E+00 sec 7.41E+00 sec 7.41E+00 sec
relative error = 0.00E+00
JKI: 3.73E+00 sec 3.73E+00 sec 3.74E+00 sec
relative error = 0.00E+00
KIJ: 1.04E+01 sec 1.03E+01 sec 1.03E+01 sec
relative error = 0.00E+00
KJI: 3.82E+00 sec 3.85E+00 sec 3.82E+00 sec
relative error = 0.00E+00
IKJ, J VECT: 9.15E+00 sec 9.14E+00 sec 9.13E+00 sec
relative error = 0.00E+00
JKI, I VECT: 2.76E+00 sec 2.76E+00 sec 2.76E+00 sec
relative error = 0.00E+00
KIJ, J VECT: 8.36E+00 sec 8.31E+00 sec 8.30E+00 sec
relative error = 0.00E+00
KJI, I VECT: 2.87E+00 sec 2.91E+00 sec 2.89E+00 sec
relative error = 0.00E+00
IJ DOT_PRODUCT: 3.46E+00 sec 3.48E+00 sec 3.48E+00 sec
relative error = 0.00E+00
JI DOT_PRODUCT: 4.90E+00 sec 4.88E+00 sec 4.87E+00 sec
relative error = 0.00E+00
IJ TP DOT_PRODUCT: 2.62E+00 sec 2.63E+00 sec 2.62E+00 sec
```

```
relative error = 0.00E+00
JI TP DOT_PRODUCT: 2.67E+00 sec 2.66E+00 sec 2.62E+00 sec
relative error = 0.00E+00
BLAS DGEMM: 8.70E-02 sec 8.70E-02 sec 9.30E-02 sec
relative error = 8.16E-16
IN BLOCKS: 2.91E+00 sec 2.75E+00 sec 2.76E+00 sec
relative error = 7.98E-16
MATMUL: 4.94E+00 sec 4.95E+00 sec 4.92E+00 sec
relative error = 0.00E+00
OWN VARIANT: 8.57E+00 sec 8.32E+00 sec 8.41E+00 sec
relative error = 1.60E-14
```

ifort N=1000, Block=100

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 6.59E-01 sec 6.63E-01 sec 6.75E-01 sec
relative error = 0.00E+00
IKJ: 6.59E-01 sec 6.59E-01 sec 6.45E-01 sec
relative error = 0.00E+00
JIK: 6.46E-01 sec 6.53E-01 sec 6.53E-01 sec
relative error = 0.00E+00
JKI: 6.57E-01 sec 6.49E-01 sec 6.61E-01 sec
relative error = 0.00E+00
KIJ: 6.59E-01 sec 6.64E-01 sec 7.04E-01 sec
relative error = 0.00E+00
KJI: 6.84E-01 sec 6.49E-01 sec 6.76E-01 sec
relative error = 0.00E+00
IKJ, J VECT: 7.66E+00 sec 7.63E+00 sec 7.61E+00 sec
relative error = 0.00E+00
JKI, I VECT: 6.42E-01 sec 6.53E-01 sec 6.53E-01 sec
relative error = 0.00E+00
KIJ, J VECT: 7.53E+00 sec 7.53E+00 sec 7.53E+00 sec
relative error = 0.00E+00
KJI, I VECT: 1.07E+00 sec 1.11E+00 sec 1.08E+00 sec
relative error = 0.00E+00
IJ DOT_PRODUCT: 1.73E+00 sec 1.72E+00 sec 1.74E+00 sec
relative error = 7.99E-16
```

```
JI DOT_PRODUCT: 6.63E-01 sec 6.55E-01 sec 6.67E-01 sec
relative error = 0.00E+00
IJ TP DOT_PRODUCT: 7.20E-01 sec 7.28E-01 sec 7.13E-01 sec
relative error = 7.99E-16
JI TP DOT_PRODUCT: 6.87E-01 sec 7.08E-01 sec 6.99E-01 sec
relative error = 7.99E-16
BLAS DGEMM: 9.67E-02 sec 7.72E-02 sec 7.48E-02 sec
relative error = 8.15E-16
IN BLOCKS: 5.81E-01 sec 3.88E-01 sec 3.90E-01 sec
relative error = 7.89E-16
MATMUL: 6.54E-01 sec 6.63E-01 sec 6.59E-01 sec
relative error = 0.00E+00
forrtl:  severe (174):  SIGSEGV, segmentation fault occurred
Image PC Routine Line Source
dmr 0000000000495C11 Unknown Unknown Unknown
dmr 0000000000493D4B Unknown Unknown Unknown
dmr 0000000000466DF4 Unknown Unknown Unknown
dmr 0000000000466C06 Unknown Unknown Unknown
dmr 00000000004453C7 Unknown Unknown Unknown
dmr 0000000000415670 Unknown Unknown Unknown
libpthread-2.27.s 00007FEFE0DA5890 Unknown Unknown Unknown
dmr 000000000041003E Unknown Unknown Unknown
dmr 000000000040F512 Unknown Unknown Unknown
dmr 0000000000403C75 Unknown Unknown Unknown
dmr 000000000040399A Unknown Unknown Unknown
dmr 0000000000402D3E Unknown Unknown Unknown
libc-2.27.so 00007FEFE09C3B97 __libc_start_main Unknown Unknown
dmr 0000000000402C2A Unknown Unknown Unknown
```

Olevel 0, N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 3.91E-03 sec 3.91E-03 sec 3.96E-03 sec
relative error = 2.98E-16
IKJ: 4.06E-03 sec 4.03E-03 sec 4.30E-03 sec
relative error = 2.98E-16
JIK: 3.96E-03 sec 3.98E-03 sec 3.99E-03 sec
```

```
relative error = 2.98E-16
JKI: 3.80E-03 sec 3.79E-03 sec 3.79E-03 sec
relative error = 2.98E-16
KIJ: 4.02E-03 sec 4.02E-03 sec 4.07E-03 sec
relative error = 2.98E-16
KJI: 3.83E-03 sec 3.83E-03 sec 3.84E-03 sec
relative error = 2.98E-16
IKJ, J VECT: 1.62E-03 sec 1.61E-03 sec 1.62E-03 sec
relative error = 2.98E-16
JKI, I VECT: 1.53E-03 sec 1.53E-03 sec 1.54E-03 sec
relative error = 2.98E-16
KIJ, J VECT: 1.62E-03 sec 1.63E-03 sec 1.61E-03 sec
relative error = 2.98E-16
KJI, I VECT: 1.54E-03 sec 1.54E-03 sec 1.53E-03 sec
relative error = 2.98E-16
IJ DOT_PRODUCT: 1.19E-03 sec 1.18E-03 sec 1.18E-03 sec
relative error = 2.98E-16
JI DOT_PRODUCT: 1.22E-03 sec 1.22E-03 sec 1.21E-03 sec
relative error = 2.98E-16
IJ TP DOT_PRODUCT: 1.21E-03 sec 1.19E-03 sec 1.18E-03 sec
relative error = 2.98E-16
JI TP DOT_PRODUCT: 1.19E-03 sec 1.18E-03 sec 1.20E-03 sec
relative error = 2.98E-16
BLAS DGEMM: 6.50E-03 sec 2.28E-04 sec 1.97E-04 sec
relative error = 2.92E-16
IN BLOCKS: 3.79E-03 sec 1.82E-03 sec 3.64E-03 sec
relative error = 1.34E-16
MATMUL: 3.30E-04 sec 2.95E-04 sec 2.94E-04 sec
relative error = 0.00E+00
OWN VARIANT: 9.98E-02 sec 6.15E-02 sec 3.91E-02 sec
relative error = 3.42E-15
```

Olevel 1, N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 3.93E-03 sec 3.91E-03 sec 3.96E-03 sec
relative error = 2.98E-16
```

```
IKJ: 4.16E-03 sec 4.10E-03 sec 4.10E-03 sec
relative error = 2.98E-16
JIK: 4.06E-03 sec 4.22E-03 sec 4.03E-03 sec
relative error = 2.98E-16
JKI: 3.80E-03 sec 3.82E-03 sec 3.87E-03 sec
relative error = 2.98E-16
KIJ: 4.06E-03 sec 4.08E-03 sec 4.03E-03 sec
relative error = 2.98E-16
KJI: 3.85E-03 sec 3.84E-03 sec 3.84E-03 sec
relative error = 2.98E-16
IKJ, J VECT: 1.62E-03 sec 1.62E-03 sec 1.61E-03 sec
relative error = 2.98E-16
JKI, I VECT: 1.52E-03 sec 1.54E-03 sec 1.53E-03 sec
relative error = 2.98E-16
KIJ, J VECT: 1.61E-03 sec 1.61E-03 sec 1.60E-03 sec
relative error = 2.98E-16
KJI, I VECT: 1.55E-03 sec 1.54E-03 sec 1.54E-03 sec
relative error = 2.98E-16
IJ DOT_PRODUCT: 1.20E-03 sec 1.20E-03 sec 1.20E-03 sec
relative error = 2.98E-16
JI DOT_PRODUCT: 1.21E-03 sec 1.22E-03 sec 1.22E-03 sec
relative error = 2.98E-16
IJ TP DOT_PRODUCT: 1.21E-03 sec 1.19E-03 sec 1.18E-03 sec
relative error = 2.98E-16
JI TP DOT_PRODUCT: 1.19E-03 sec 1.19E-03 sec 1.18E-03 sec
relative error = 2.98E-16
BLAS DGEMM: 8.77E-03 sec 2.45E-04 sec 1.99E-04 sec
relative error = 2.92E-16
IN BLOCKS: 4.97E-03 sec 1.35E-02 sec 7.27E-03 sec
relative error = 1.34E-16
MATMUL: 3.18E-04 sec 2.91E-04 sec 2.88E-04 sec
relative error = 0.00E+00
OWN VARIANT: 7.09E-02 sec 1.02E-01 sec 3.01E-02 sec
relative error = 3.42E-15
```

Olevel 2, N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
```

```
TIMING RESULTS:
IJK: 3.90E-03 sec 3.90E-03 sec 3.94E-03 sec
relative error = 2.98E-16
IKJ: 4.09E-03 sec 4.08E-03 sec 4.10E-03 sec
relative error = 2.98E-16
JIK: 3.88E-03 sec 3.85E-03 sec 3.86E-03 sec
relative error = 2.98E-16
JKI: 3.79E-03 sec 3.85E-03 sec 3.80E-03 sec
relative error = 2.98E-16
KIJ: 3.99E-03 sec 3.98E-03 sec 4.07E-03 sec
relative error = 2.98E-16
KJI: 3.83E-03 sec 3.84E-03 sec 3.82E-03 sec
relative error = 2.98E-16
IKJ, J VECT: 1.62E-03 sec 1.61E-03 sec 1.62E-03 sec
relative error = 2.98E-16
JKI, I VECT: 1.53E-03 sec 1.52E-03 sec 1.54E-03 sec
relative error = 2.98E-16
KIJ, J VECT: 1.61E-03 sec 1.60E-03 sec 1.59E-03 sec
relative error = 2.98E-16
KJI, I VECT: 1.53E-03 sec 1.54E-03 sec 1.53E-03 sec
relative error = 2.98E-16
IJ DOT_PRODUCT: 1.19E-03 sec 1.20E-03 sec 1.19E-03 sec
relative error = 2.98E-16
JI DOT_PRODUCT: 1.20E-03 sec 1.22E-03 sec 1.22E-03 sec
relative error = 2.98E-16
IJ TP DOT_PRODUCT: 1.20E-03 sec 1.19E-03 sec 1.19E-03 sec
relative error = 2.98E-16
JI TP DOT_PRODUCT: 1.19E-03 sec 1.19E-03 sec 1.19E-03 sec
relative error = 2.98E-16
BLAS DGEMM: 1.07E-02 sec 2.38E-04 sec 1.95E-04 sec
relative error = 2.92E-16
IN BLOCKS: 4.93E-03 sec 4.96E-03 sec 4.95E-03 sec
relative error = 1.34E-16
MATMUL: 3.13E-04 sec 2.90E-04 sec 2.90E-04 sec
relative error = 0.00E+00
OWN VARIANT: 8.22E-02 sec 9.35E-02 sec 3.43E-02 sec
relative error = 3.42E-15
```

Olevel 3, N=100, Block=10

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 3.90E-03 sec 3.92E-03 sec 4.02E-03 sec
relative error = 2.98E-16
IKJ: 4.08E-03 sec 4.08E-03 sec 4.22E-03 sec
relative error = 2.98E-16
JIK: 4.11E-03 sec 3.96E-03 sec 3.97E-03 sec
relative error = 2.98E-16
JKI: 3.81E-03 sec 3.79E-03 sec 3.82E-03 sec
relative error = 2.98E-16
KIJ: 4.06E-03 sec 4.06E-03 sec 4.04E-03 sec
relative error = 2.98E-16
KJI: 3.83E-03 sec 3.83E-03 sec 3.86E-03 sec
relative error = 2.98E-16
IKJ, J VECT: 1.69E-03 sec 1.66E-03 sec 1.66E-03 sec
relative error = 2.98E-16
JKI, I VECT: 1.54E-03 sec 1.54E-03 sec 1.56E-03 sec
relative error = 2.98E-16
KIJ, J VECT: 1.64E-03 sec 1.65E-03 sec 1.64E-03 sec
relative error = 2.98E-16
KJI, I VECT: 1.56E-03 sec 1.57E-03 sec 1.56E-03 sec
relative error = 2.98E-16
IJ DOT_PRODUCT: 1.20E-03 sec 1.19E-03 sec 1.19E-03 sec
relative error = 2.98E-16
JI DOT_PRODUCT: 1.21E-03 sec 1.22E-03 sec 1.21E-03 sec
relative error = 2.98E-16
IJ TP DOT_PRODUCT: 1.21E-03 sec 1.20E-03 sec 1.19E-03 sec
relative error = 2.98E-16
JI TP DOT_PRODUCT: 1.18E-03 sec 1.19E-03 sec 1.19E-03 sec
relative error = 2.98E-16
BLAS DGEMM: 8.44E-03 sec 2.13E-04 sec 2.14E-04 sec
relative error = 2.92E-16
IN BLOCKS: 4.27E-03 sec 3.99E-03 sec 1.71E-02 sec
relative error = 1.34E-16
MATMUL: 3.36E-04 sec 3.06E-04 sec 3.07E-04 sec
relative error = 0.00E+00
```

```
OWN VARIANT: 9.54E-02 sec 7.15E-02 sec 3.31E-02 sec
relative error = 3.42E-15
```

Olevel 0, N=500, Block=50

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 5.67E-01 sec 5.68E-01 sec 5.69E-01 sec
relative error = 5.75E-16
IKJ: 1.12E+00 sec 1.11E+00 sec 1.11E+00 sec
relative error = 5.75E-16
JIK: 9.90E-01 sec 1.01E+00 sec 1.02E+00 sec
relative error = 5.75E-16
JKI: 4.66E-01 sec 4.66E-01 sec 4.66E-01 sec
relative error = 5.75E-16
KIJ: 1.10E+00 sec 1.11E+00 sec 1.11E+00 sec
relative error = 5.75E-16
KJI: 4.81E-01 sec 4.80E-01 sec 4.81E-01 sec
relative error = 5.75E-16
IKJ, J VECT: 4.28E-01 sec 4.27E-01 sec 4.28E-01 sec
relative error = 5.75E-16
JKI, I VECT: 1.89E-01 sec 1.88E-01 sec 1.88E-01 sec
relative error = 5.75E-16
KIJ, J VECT: 4.24E-01 sec 4.24E-01 sec 4.23E-01 sec
relative error = 5.75E-16
KJI, I VECT: 1.98E-01 sec 1.97E-01 sec 1.96E-01 sec
relative error = 5.75E-16
IJ DOT_PRODUCT: 1.63E-01 sec 1.63E-01 sec 1.64E-01 sec
relative error = 5.75E-16
JI DOT_PRODUCT: 2.74E-01 sec 2.68E-01 sec 2.59E-01 sec
relative error = 5.75E-16
IJ TP DOT_PRODUCT: 1.50E-01 sec 1.51E-01 sec 1.51E-01 sec
relative error = 5.75E-16
JI TP DOT_PRODUCT: 1.50E-01 sec 1.50E-01 sec 1.50E-01 sec
relative error = 5.75E-16
BLAS DGEMM: 2.79E-02 sec 1.19E-02 sec 1.50E-02 sec
relative error = 3.13E-16
IN BLOCKS: 3.18E-01 sec 1.53E-01 sec 1.53E-01 sec
```

```
relative error = 1.40E-16
MATMUL: 1.04E-02 sec 1.04E-02 sec 1.04E-02 sec
relative error = 0.00E+00
OWN VARIANT: 7.28E-01 sec 7.31E-01 sec 7.30E-01 sec
relative error = 1.38E-14
```

Olevel 1, N=500, Block=50

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 5.67E-01 sec 5.68E-01 sec 5.69E-01 sec
relative error = 5.75E-16
IKJ: 1.11E+00 sec 1.12E+00 sec 1.12E+00 sec
relative error = 5.75E-16
JIK: 1.01E+00 sec 1.02E+00 sec 1.01E+00 sec
relative error = 5.75E-16
JKI: 4.66E-01 sec 4.66E-01 sec 4.66E-01 sec
relative error = 5.75E-16
KIJ: 1.11E+00 sec 1.12E+00 sec 1.11E+00 sec
relative error = 5.75E-16
KJI: 4.81E-01 sec 4.81E-01 sec 4.81E-01 sec
relative error = 5.75E-16
IKJ, J VECT: 4.25E-01 sec 4.31E-01 sec 4.30E-01 sec
relative error = 5.75E-16
JKI, I VECT: 1.88E-01 sec 1.90E-01 sec 1.88E-01 sec
relative error = 5.75E-16
KIJ, J VECT: 4.21E-01 sec 4.20E-01 sec 4.08E-01 sec
relative error = 5.75E-16
KJI, I VECT: 1.97E-01 sec 1.96E-01 sec 1.98E-01 sec
relative error = 5.75E-16
IJ DOT_PRODUCT: 1.65E-01 sec 1.65E-01 sec 1.66E-01 sec
relative error = 5.75E-16
JI DOT_PRODUCT: 2.73E-01 sec 2.73E-01 sec 2.68E-01 sec
relative error = 5.75E-16
IJ TP DOT_PRODUCT: 1.51E-01 sec 1.51E-01 sec 1.51E-01 sec
relative error = 5.75E-16
JI TP DOT_PRODUCT: 1.52E-01 sec 1.51E-01 sec 1.51E-01 sec
relative error = 5.75E-16
```

```
BLAS DGEMM: 8.72E-03 sec 2.60E-02 sec 1.45E-02 sec
relative error = 3.13E-16
IN BLOCKS: 3.58E-01 sec 1.52E-01 sec 1.52E-01 sec
relative error = 1.40E-16
MATMUL: 1.06E-02 sec 1.04E-02 sec 1.05E-02 sec
relative error = 0.00E+00
OWN VARIANT: 7.32E-01 sec 7.25E-01 sec 7.27E-01 sec
relative error = 1.38E-14
```

Olevel 2, N=500, Block=50

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 5.62E-01 sec 5.66E-01 sec 5.70E-01 sec
relative error = 5.75E-16
IKJ: 1.13E+00 sec 1.14E+00 sec 1.14E+00 sec
relative error = 5.75E-16
JIK: 1.05E+00 sec 1.04E+00 sec 1.05E+00 sec
relative error = 5.75E-16
JKI: 4.69E-01 sec 4.67E-01 sec 4.67E-01 sec
relative error = 5.75E-16
KIJ: 1.13E+00 sec 1.12E+00 sec 1.14E+00 sec
relative error = 5.75E-16
KJI: 4.81E-01 sec 4.84E-01 sec 4.82E-01 sec
relative error = 5.75E-16
IKJ, J VECT: 4.21E-01 sec 4.32E-01 sec 4.34E-01 sec
relative error = 5.75E-16
JKI, I VECT: 1.89E-01 sec 1.89E-01 sec 1.89E-01 sec
relative error = 5.75E-16
KIJ, J VECT: 4.28E-01 sec 4.26E-01 sec 4.24E-01 sec
relative error = 5.75E-16
KJI, I VECT: 1.96E-01 sec 1.96E-01 sec 1.98E-01 sec
relative error = 5.75E-16
IJ DOT_PRODUCT: 1.65E-01 sec 1.65E-01 sec 1.64E-01 sec
relative error = 5.75E-16
JI DOT_PRODUCT: 2.50E-01 sec 2.62E-01 sec 2.64E-01 sec
relative error = 5.75E-16
IJ TP DOT_PRODUCT: 1.51E-01 sec 1.53E-01 sec 1.52E-01 sec
```

```
relative error = 5.75E-16
JI TP DOT_PRODUCT: 1.51E-01 sec 1.51E-01 sec 1.51E-01 sec
relative error = 5.75E-16
BLAS DGEMM: 2.98E-02 sec 1.11E-02 sec 1.10E-02 sec
relative error = 3.13E-16
IN BLOCKS: 2.78E-01 sec 1.51E-01 sec 1.51E-01 sec
relative error = 1.40E-16
MATMUL: 1.05E-02 sec 1.04E-02 sec 1.04E-02 sec
relative error = 0.00E+00
OWN VARIANT: 7.32E-01 sec 7.30E-01 sec 7.29E-01 sec
relative error = 1.38E-14
```

Olevel 3, N=500, Block=50

```
 Enter the value for N: Enter the blocksize of the sub-blocks:
TIMING RESULTS:
IJK: 5.69E-01 sec 5.68E-01 sec 5.63E-01 sec
relative error = 5.75E-16
IKJ: 1.12E+00 sec 1.11E+00 sec 1.12E+00 sec
relative error = 5.75E-16
JIK: 1.02E+00 sec 1.03E+00 sec 1.03E+00 sec
relative error = 5.75E-16
JKI: 4.65E-01 sec 4.67E-01 sec 4.66E-01 sec
relative error = 5.75E-16
KIJ: 1.08E+00 sec 1.10E+00 sec 1.08E+00 sec
relative error = 5.75E-16
KJI: 4.79E-01 sec 4.79E-01 sec 4.82E-01 sec
relative error = 5.75E-16
IKJ, J VECT: 4.05E-01 sec 4.00E-01 sec 4.12E-01 sec
relative error = 5.75E-16
JKI, I VECT: 1.87E-01 sec 1.87E-01 sec 1.86E-01 sec
relative error = 5.75E-16
KIJ, J VECT: 3.85E-01 sec 3.92E-01 sec 4.14E-01 sec
relative error = 5.75E-16
KJI, I VECT: 1.94E-01 sec 1.95E-01 sec 1.97E-01 sec
relative error = 5.75E-16
IJ DOT_PRODUCT: 1.66E-01 sec 1.66E-01 sec 1.65E-01 sec
relative error = 5.75E-16
```

```
JI DOT_PRODUCT: 2.69E-01 sec 2.68E-01 sec 2.72E-01 sec
relative error = 5.75E-16
IJ TP DOT_PRODUCT: 1.52E-01 sec 1.53E-01 sec 1.52E-01 sec
relative error = 5.75E-16
JI TP DOT_PRODUCT: 1.50E-01 sec 1.50E-01 sec 1.50E-01 sec
relative error = 5.75E-16
BLAS DGEMM: 1.92E-02 sec 1.28E-02 sec 1.22E-02 sec
relative error = 3.13E-16
IN BLOCKS: 3.34E-01 sec 1.50E-01 sec 1.50E-01 sec
relative error = 1.40E-16
MATMUL: 1.04E-02 sec 1.05E-02 sec 1.04E-02 sec
relative error = 0.00E+00
OWN VARIANT: 7.28E-01 sec 7.28E-01 sec 7.27E-01 sec
relative error = 1.38E-14
```

Test of block sizes

```
Enter the value for N: 1386
Enter the blocksize of the sub-blocks:  22
TIMING RESULTS:
IN BLOCKS: 2.01E+00 sec 2.04E+00 sec 2.03E+00 sec
relative error = 2.61E-16
Enter the value for N: 1386
Enter the blocksize of the sub-blocks:  63
TIMING RESULTS:
IN BLOCKS: 2.32E+00 sec 2.31E+00 sec 2.31E+00 sec
relative error = 1.88E-16
```

test.dat: contains execution times,not MFlop/s

```
 22 1.60E-05 1.40E-05 1.60E-05 2.00E-05 1.60E-05 1.60E-05 7.40E-05 6.00E-06 4.00E-06
2.00E-05 1.60E-05 1.80E-05 1.60E-05 1.40E-05 1.40E-05
44 1.28E-04 1.36E-04 1.34E-04 1.76E-04 1.56E-04 1.72E-04 1.12E-04 3.20E-05 2.80E-05
1.64E-04 2.38E-04 1.30E-04 2.80E-05 2.40E-05 2.20E-05
66 4.06E-04 5.02E-04 5.00E-04 7.18E-04 6.94E-04 6.92E-04 6.29E-03 6.60E-05 5.60E-05
6.26E-04 6.41E-04 4.68E-04 7.80E-05 7.40E-05 7.30E-05
88 9.02E-04 8.81E-04 8.85E-04 1.22E-03 1.20E-03 1.20E-03 1.11E-03 1.33E-04 1.28E-04
1.49E-03 1.49E-03 1.48E-03 2.32E-04 2.13E-04 2.13E-04
110 2.55E-03 2.50E-03 2.23E-03 2.67E-03 2.54E-03 4.33E-03 5.55E-03 3.05E-04 2.19E-04
2.90E-03 2.90E-03 2.90E-03 4.60E-04 4.06E-04 4.06E-04
```

```
132 4.23E-03 4.21E-03 4.23E-03 6.65E-03 6.65E-03 6.85E-03 7.39E-03 4.15E-04 2.68E-04
4.99E-03 3.65E-03 3.85E-03 6.72E-04 6.31E-04 6.27E-04

154 6.75E-03 6.65E-03 6.62E-03 7.64E-03 8.94E-03 1.89E-02 6.18E-04 4.33E-04 4.29E-04
7.92E-03 7.95E-03 7.58E-03 7.51E-04 6.90E-04 6.83E-04

176 9.99E-03 9.83E-03 9.82E-03 2.20E-02 1.65E-02 1.64E-02 8.33E-04 6.18E-04 6.04E-04
5.78E-03 1.04E-02 1.19E-02 1.49E-03 9.55E-03 1.48E-03

198 1.42E-02 1.29E-02 9.70E-03 3.20E-02 2.42E-02 2.07E-02 5.32E-03 9.51E-04 7.46E-04
1.41E-02 1.52E-02 1.69E-02 2.20E-03 2.05E-03 2.03E-03

220 1.93E-02 1.58E-02 1.65E-02 4.11E-02 2.73E-02 2.56E-02 5.62E-03 1.22E-03 1.09E-03
2.32E-02 2.85E-02 2.02E-02 2.75E-03 2.63E-03 2.62E-03

242 2.53E-02 3.03E-02 2.32E-02 4.23E-02 1.56E-02 1.49E-02 1.09E-02 1.68E-03 1.42E-03
3.08E-02 3.64E-02 4.07E-02 4.93E-03 4.80E-03 4.79E-03

264 4.71E-02 4.29E-02 4.07E-02 2.15E-02 1.95E-02 1.94E-02 2.60E-02 1.73E-03 1.16E-03
5.05E-02 5.28E-02 5.39E-02 6.04E-03 6.40E-03 6.19E-03

286 5.47E-02 4.51E-02 1.42E-02 2.50E-02 2.50E-02 2.48E-02 1.95E-03 1.28E-02 1.71E-03
7.20E-02 1.06E-01 6.18E-02 2.43E-03 2.31E-03 2.41E-03

308 1.77E-02 1.73E-02 1.74E-02 3.13E-02 3.14E-02 3.11E-02 8.55E-03 2.94E-03 5.82E-03
5.93E-02 6.33E-02 7.14E-02 5.58E-03 6.16E-03 7.59E-03

330 2.18E-02 2.15E-02 2.15E-02 3.88E-02 3.91E-02 3.86E-02 3.60E-03 3.15E-03 7.32E-03
8.27E-02 7.51E-02 6.65E-02 3.47E-03 3.40E-03 3.50E-03

352 2.62E-02 2.60E-02 2.61E-02 4.73E-02 4.73E-02 4.74E-02 3.99E-03 8.68E-03 3.52E-03
9.42E-02 9.91E-02 4.53E-02 3.96E-03 3.89E-03 3.90E-03

374 3.23E-02 3.14E-02 3.14E-02 5.72E-02 5.76E-02 5.72E-02 1.46E-02 3.83E-03 3.60E-03
1.41E-01 6.53E-02 3.99E-02 4.92E-03 4.83E-03 4.83E-03

396 3.82E-02 3.80E-02 3.78E-02 6.87E-02 6.88E-02 6.85E-02 2.89E-02 6.17E-03 5.96E-03
1.74E-01 5.02E-02 4.71E-02 5.46E-03 5.48E-03 5.40E-03

418 4.48E-02 4.43E-02 4.43E-02 8.12E-02 8.09E-02 8.06E-02 2.70E-02 4.63E-03 4.51E-03
1.74E-01 5.58E-02 5.57E-02 6.69E-03 6.57E-03 6.64E-03

440 5.51E-02 5.44E-02 5.46E-02 9.55E-02 9.54E-02 9.52E-02 2.99E-02 8.78E-03 8.80E-03
1.92E-01 7.26E-02 6.44E-02 7.42E-03 7.27E-03 7.29E-03

462 6.25E-02 6.17E-02 6.17E-02 1.11E-01 1.11E-01 1.10E-01 1.49E-02 8.57E-03 8.13E-03
2.18E-01 7.48E-02 7.48E-02 8.57E-03 8.52E-03 8.53E-03

484 7.36E-02 7.26E-02 7.26E-02 1.29E-01 1.29E-01 1.29E-01 3.32E-02 1.71E-02 1.64E-02
2.32E-01 8.60E-02 8.59E-02 9.49E-03 9.51E-03 9.56E-03

506 8.49E-02 8.47E-02 8.69E-02 1.49E-01 1.49E-01 1.49E-01 3.52E-02 1.31E-02 1.19E-02
2.22E-01 9.99E-02 9.96E-02 1.12E-02 1.11E-02 1.11E-02

528 1.03E-01 1.01E-01 1.01E-01 1.71E-01 1.70E-01 1.69E-01 1.79E-02 1.30E-02 1.23E-02
2.63E-01 1.13E-01 1.11E-01 1.36E-02 1.36E-02 1.36E-02

550 1.16E-01 1.15E-01 1.16E-01 1.94E-01 1.94E-01 1.93E-01 2.87E-02 1.35E-02 1.71E-02
2.63E-01 1.26E-01 1.26E-01 1.55E-02 1.53E-02 1.54E-02

572 1.33E-01 1.32E-01 1.35E-01 2.20E-01 2.21E-01 2.19E-01 2.76E-02 1.66E-02 1.67E-02
2.97E-01 1.42E-01 1.42E-01 1.70E-02 1.70E-02 1.70E-02

594 1.55E-01 1.53E-01 1.55E-01 2.47E-01 2.47E-01 2.47E-01 2.98E-02 2.25E-02 1.81E-02
3.36E-01 1.60E-01 1.60E-01 1.91E-02 1.90E-02 1.92E-02
```

```
616 1.73E-01 1.70E-01 1.71E-01 2.76E-01 2.76E-01 2.75E-01 2.81E-02 1.95E-02 1.99E-02
3.54E-01 1.78E-01 1.79E-01 2.07E-02 2.06E-02 2.06E-02

638 1.97E-01 1.94E-01 1.96E-01 3.08E-01 3.08E-01 3.08E-01 1.06E-01 2.07E-02 3.37E-02
3.72E-01 2.00E-01 1.99E-01 2.29E-02 2.29E-02 2.29E-02

660 2.18E-01 2.23E-01 2.23E-01 3.42E-01 3.41E-01 3.39E-01 3.50E-02 2.67E-02 2.77E-02
4.17E-01 2.20E-01 2.20E-01 2.51E-02 2.50E-02 2.50E-02

682 2.48E-01 2.48E-01 2.47E-01 3.78E-01 3.75E-01 3.77E-01 1.40E-01 2.70E-02 2.79E-02
4.15E-01 2.46E-01 2.46E-01 2.76E-02 2.76E-02 2.76E-02

704 2.71E-01 2.75E-01 2.76E-01 4.16E-01 4.13E-01 4.16E-01 1.13E-01 1.84E-02 1.85E-02
4.64E-01 2.67E-01 2.67E-01 2.99E-02 2.98E-02 2.99E-02

726 3.08E-01 3.06E-01 3.07E-01 4.58E-01 4.57E-01 4.54E-01 3.28E-02 3.23E-02 2.97E-02
4.53E-01 2.92E-01 2.90E-01 3.29E-02 3.29E-02 3.27E-02

748 3.36E-01 3.38E-01 3.35E-01 5.01E-01 5.00E-01 5.00E-01 4.34E-02 3.63E-02 3.63E-02
5.15E-01 3.19E-01 3.18E-01 3.55E-02 3.54E-02 3.54E-02

770 3.65E-01 3.64E-01 3.64E-01 5.47E-01 5.45E-01 5.45E-01 4.02E-02 2.50E-02 3.41E-02
5.49E-01 3.49E-01 3.49E-01 4.06E-02 4.06E-02 4.06E-02

792 4.01E-01 4.02E-01 4.00E-01 5.93E-01 5.94E-01 5.93E-01 3.55E-02 2.94E-02 3.82E-02
6.08E-01 3.76E-01 3.76E-01 4.35E-02 4.34E-02 4.34E-02

814 4.36E-01 4.36E-01 4.36E-01 6.46E-01 6.44E-01 6.47E-01 8.61E-02 3.22E-02 3.02E-02
6.07E-01 4.11E-01 4.10E-01 4.74E-02 4.75E-02 4.73E-02

836 4.67E-01 4.70E-01 4.70E-01 7.00E-01 6.99E-01 6.98E-01 4.75E-02 4.76E-02 4.23E-02
6.21E-01 4.44E-01 4.45E-01 5.05E-02 5.05E-02 5.06E-02

858 5.05E-01 5.09E-01 5.09E-01 7.57E-01 7.55E-01 7.55E-01 5.30E-02 5.56E-02 4.79E-02
6.73E-01 4.80E-01 4.79E-01 5.50E-02 5.51E-02 5.51E-02

880 5.54E-01 5.50E-01 5.42E-01 8.20E-01 8.19E-01 8.34E-01 9.83E-02 4.88E-02 5.03E-02
7.17E-01 5.14E-01 5.18E-01 5.78E-02 5.77E-02 5.76E-02
```