Cris Liao
862396679
CS211

HW2

The order of the solution is Q1, Q3, then Q2.

1. For a non-pivoting and non-block version.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 13 & 18 \\ 7 & 54 & 78 \end{bmatrix}$$ Original matrix

$$= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 40 & 57 \end{bmatrix}$$ First , we calculate first column: $\frac{4}{1} = 4, \frac{7}{1} = 7$

Update the small 2*2 matrix: $13 - 4 * 2 = 5$,
$18 - 4 * 3 = 6, 54 - 7 * 2 = 40, 78 - 7 * 3 = 57$

$$= \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$ Update the first column second row of small 2*2 matrix:
$\frac{40}{5} = 8$

Update the last element after previous step:
$57 - 8 * 6 = 9$

So the L is left bottom triangle, and U is right top triangle, which is

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 8 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 0 & 9 \end{bmatrix}$$

Checking: If we multiply both matrix, A= L*U:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 8 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 0 & 9 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 13 & 18 \\ 7 & 54 & 78 \end{bmatrix}$$

3.

For blocked and non-pivoting version

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 9 & 12 & 15 \\ 3 & 26 & 41 & 49 \\ 5 & 40 & 107 & 135 \end{bmatrix}$$

Original Matrix

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 9 & 12 & 15 \\ 3 & 26 & 41 & 49 \\ 5 & 40 & 107 & 135 \end{bmatrix}$$

$\frac{2}{1} = 2$

$\frac{3}{1} = 3$

$\frac{5}{1} = 5$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 12 & 15 \\ 3 & 20 & 41 & 49 \\ 5 & 30 & 107 & 135 \end{bmatrix}$$

$9 - 2 * 2 = 5$

$26 - 2 * 3 = 20$

$40 - 5 * 2 = 30$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 12 & 15 \\ 3 & 4 & 41 & 49 \\ 5 & 6 & 107 & 135 \end{bmatrix}$$

$\frac{20}{5} = 4$

$\frac{30}{5} = 6$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 6 & 7 \\ 3 & 4 & 41 & 49 \\ 5 & 6 & 107 & 135 \end{bmatrix}$$

Calculate delay update now:

$12 - 2 * 3 = 6$

$15 - 2 * 4 = 7$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 6 & 7 \\ 3 & 4 & 8 & 9 \\ 5 & 6 & 56 & 73 \end{bmatrix}$$

Calculate the right corner small 2*2 matrix:

$$\begin{bmatrix} 41 & 49 \\ 107 & 135 \end{bmatrix} - \begin{bmatrix} 3 & 4 \\ 5 & 6 \end{bmatrix} * \begin{bmatrix} 3 & 4 \\ 6 & 7 \end{bmatrix} = \begin{bmatrix} 8 & 9 \\ 56 & 73 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 6 & 7 \\ 3 & 4 & 8 & 9 \\ 5 & 6 & 7 & 10 \end{bmatrix}$$

Calculate last 2*2 matrix by using same method:

$\frac{56}{8} = 7$

$73 - 7 * 9 = 10$

Verify answers:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ 5 & 6 & 7 & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 5 & 6 & 7 \\ 0 & 0 & 8 & 9 \\ 0 & 0 & 0 & 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ 5 & 6 & 7 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 5 & 6 & 7 \\ 0 & 0 & 8 & 9 \\ 0 & 0 & 0 & 10 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 9 & 12 & 15 \\ 3 & 26 & 41 & 49 \\ 5 & 40 & 107 & 135 \end{bmatrix}$$

2. Below is the screenshot of the runtime for both my version and lapack version.

My version:

```
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main my 1000
n=1000, pad=1
time=0.136194s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main my 2000
n=2000, pad=1
time=1.671660s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main my 3000
n=3000, pad=1
time=7.236257s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main my 4000
n=4000, pad=1
time=16.406926s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main my 5000
n=5000, pad=1
time=30.592321s
```

Lackpack version:

```
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main lapack 1000
n=1000, pad=1
time=0.062928s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main lapack 2000
n=2000, pad=1
time=0.250752s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main lapack 3000
n=3000, pad=1
time=0.693564s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main lapack 4000
n=4000, pad=1
time=1.411913s
[eliao014@cluster-001-login-node cs211-hw2-solving-large-linear-system-CRIS66666]$ ./main lapack 5000
n=5000, pad=1
time=2.869858s
```

Gflops estimate for my version of code:

LU decomposition: $\frac{2}{3}n^3$

Forward substitution: $n^2$

Backward substitution: $n^2$

Total flops: $\frac{2}{3}n^3 + 2n^2$

For n=1000, we have $\frac{2}{3} * 1000^3 + 2 * 1000^2$=668666666.7 flops

Gflops=$\frac{Total\ floating-point\ operations}{runtime*10^9} = \frac{668666666.7}{0.136194*10^9} = 4.91$

Gflops estimate for lapack version of code:

Total flops: $\frac{2}{3}n^3$

For n=1000, we have $\frac{2}{3} * 1000^3$ =666666666.7 flops

Gflops=$\frac{Total\ floating-point\ operations}{runtime*10^9} = \frac{668666666.7}{0.062928*10^9} = 10.59$

| | n=1000 | n=2000 | n=3000 | n=4000 | n=5000 |
|---|---|---|---|---|---|
| My-runtime (s) | 0.136194 | 1.671660 | 7.236257 | 16.406926 | 30.592321 |
| my-Gflops | 4.91 | 3.20 | 2.49 | 2.60 | 2.73 |
| Lapack-runtime (s) | 0.062928 | 0.250752 | 0.693564 | 1.411913 | 2.869858 |
| lapack-Gflops | 10.59 | 21.28 | 25.95 | 30.21 | 29.04 |