

Hw2 Hairu Wen

GitHub Username: hwen020

<https://github.com/CS211-Fall2023/hw2-hwen020>

SID: 862467599

1.

suppose $EA=U$.
where $E=E_{32} \cdot E_{31} \cdot E_{21}$.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 13 & 18 \\ 7 & 54 & 78 \end{pmatrix} \xrightarrow{r_2 - 4r_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 7 & 54 & 78 \end{pmatrix} \xrightarrow{r_3 - 7r_1} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 40 & 57 \end{pmatrix} \xrightarrow{r_3 - 8r_2} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 0 & 9 \end{pmatrix}$$

$$E_{32} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -8 & 1 \end{pmatrix} \quad E_{31} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -7 & 0 & 1 \end{pmatrix} \quad E_{21} = \begin{pmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$E = E_{32} E_{31} E_{21} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -8 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -7 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -7 & -8 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 25 & -8 & 1 \end{pmatrix}$$

So

$$A = E^{-1}U, (E|I) = \begin{pmatrix} 1 & 0 & 0 & | & 1 & 0 & 0 \\ -4 & 1 & 0 & | & 0 & 1 & 0 \\ 25 & -8 & 1 & | & 0 & 0 & 1 \end{pmatrix} \xrightarrow{r_2 + 4r_1} \begin{pmatrix} 1 & 0 & 0 & | & 1 & 0 & 0 \\ 0 & 1 & 0 & | & 4 & 1 & 0 \\ 0 & -8 & 1 & | & -25 & 0 & 1 \end{pmatrix} \xrightarrow{r_3 + 8r_2} \begin{pmatrix} 1 & 0 & 0 & | & 1 & 0 & 0 \\ 0 & 1 & 0 & | & 4 & 1 & 0 \\ 0 & 0 & 1 & | & 7 & 8 & 1 \end{pmatrix}$$
$$\therefore E^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 8 & 1 \end{pmatrix} \quad \therefore A = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 8 & 1 \end{pmatrix}}_L \underbrace{\begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 0 & 9 \end{pmatrix}}_U$$

2. Test: implementation and LAPACK version with matrix size 1000, 2000, 3000, 4000, 5000.
Compare the performance (i.e., Gflops) of the two approaches.

```

[hw020@cluster-001-login-node hw2-hw020]$ ./main my 1000
n=1000, pad=1
time=0.137454s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 2000
n=2000, pad=1
time=1.824548s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 3000
n=3000, pad=1
time=6.281816s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 4000
n=4000, pad=1
time=16.161094s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 5000
n=5000, pad=1
time=33.633014s
[hw020@cluster-001-login-node hw2-hw020]$ ./main lapack 1000
n=1000, pad=1
time=0.078519s
[hw020@cluster-001-login-node hw2-hw020]$ ./main lapack 2000
n=2000, pad=1
time=0.266170s
[hw020@cluster-001-login-node hw2-hw020]$ ./main lapack 3000
n=3000, pad=1
time=0.775780s
[hw020@cluster-001-login-node hw2-hw020]$ ./main lapack 4000
n=4000, pad=1
time=1.503223s
[hw020@cluster-001-login-node hw2-hw020]$ ./main lapack 5000
n=5000, pad=1
time=3.041906s

```

```

[hw020@cluster-001-login-node hw2-hw020]$ ./main my 1000
n=1000, pad=1
time=0.137454s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 2000
n=2000, pad=1
time=1.824548s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 3000
n=3000, pad=1
time=6.281816s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 4000
n=4000, pad=1
time=16.161094s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my 5000
n=5000, pad=1
time=33.633014s

```

```
Gflops.py 1 def gflops(n, time):
main.py 2 GF = 2 * n * n * n / (time * 1e9)
ernal Librarie 3 print(f'{GF} Gflops')
atches and Co 4
5 gflops(1000, 0.137454)
6 gflops(2000, 1.824548)
7 gflops(3000, 6.281816)
8 gflops(4000, 16.161094)
9 gflops(5000, 33.633014)

Gflops x
C:\Users\12541\AppData\Local\Programs\Python\Python38\p
14.550322289638716 Gflops
8.769295189822357 Gflops
8.596240322862052 Gflops
7.920255893567602 Gflops
7.433172655891023 Gflops
```

```
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main lapack 1000
n=1000, pad=1
time=0.078519s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main lapack 2000
n=2000, pad=1
time=0.266170s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main lapack 3000
n=3000, pad=1
time=0.775780s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main lapack 4000
n=4000, pad=1
time=1.503223s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main lapack 5000
n=5000, pad=1
time=3.041906s
```

```

thonProject ( 1  def gflops(n, time):
Gflops.py      2      GF = 2 * n * n * n / (time * 1e9)
main.py        3      print(f'{GF} Gflops')
External Libraries 4
atches and Co 5
6      gflops(1000, 0.078519)
7      gflops(2000, 0.266170)
8      gflops(3000, 0.775780)
9      gflops(4000, 1.503223)
10     gflops(5000, 3.041906)

Gflops x
C:\Users\12541\AppData\Local\Programs\Python\Python38\
25.471541919790113 Gflops
60.111958522748616 Gflops
69.60736291216583 Gflops
85.15037356400215 Gflops
82.18531407610887 Gflops

```

We can see from above that lapack has a better performance on this task.

3.

Handwritten notes showing matrix operations and calculations:

① 1st column $\rightarrow \frac{a_{i1}}{a_{11}}$ else remain the same

② 2nd column $\rightarrow a_{22} = 9 - 2 \times 2 = 5$
 $a_{32} = 26 - 3 \times 2 = 20$
 $a_{42} = 40 - 5 \times 2 = 30$

③ $a_{32} = \frac{a_{32}}{a_{22}} = 4$
 $\rightarrow a_{42} = \frac{a_{42}}{a_{22}} = 6$

④ $a_{33} = a_{33} - a_{21}a_{13}$
 $= 12 - 6 = 6$
 $a_{24} = a_{24} - a_{21}a_{14}$
 $= 15 - 2 \times 4 = 7$

Matrices shown:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 9 & 12 & 15 \\ 3 & 26 & 41 & 49 \\ 5 & 40 & 107 & 135 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & & & \\ 2 & & & \\ 3 & & & \\ 5 & & & \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 2 & & \\ 2 & 5 & & \\ 3 & 20 & & \\ 5 & 30 & & \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & & \\ 2 & 5 & & \\ 3 & 4 & & \\ 5 & 6 & & \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 5 & 6 & 7 \\ 3 & 4 & & \\ 5 & 6 & & \end{pmatrix}$$

$$\begin{aligned}
 \textcircled{5} \quad a_{33} &= a_{33} - a_{13}a_{31} - a_{23}a_{32} = 41 - 9 - 24 = 8 \\
 \rightarrow a_{34} &= a_{34} - a_{31}a_{14} - a_{32}a_{24} = 49 - 3 \times 4 - 4 \times 7 = 9 \\
 a_{43} &= a_{43} - a_{41}a_{13} - a_{42}a_{23} = 107 - 3 \times 5 - 6 \times 6 = 56 \\
 &\quad 107 - 15 - 36 = 56 \\
 a_{44} &= a_{44} - a_{41}a_{14} - a_{42}a_{24} = 135 - 4 \times 5 - 6 \times 7 \\
 &\quad = 135 - 20 - 42 = 73
 \end{aligned}$$

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

$$\begin{aligned}
 \textcircled{6} \quad a_{43} &= a_{43} - a_{33} = 7 & \textcircled{7} \quad a_{44} &= a_{44} - a_{43}a_{34} \\
 \rightarrow \cancel{a_{44} - a_{41}a_{14} - a_{42}a_{24}} &= 73 - 7 \times 9 = 10
 \end{aligned}$$

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

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

Bonus Point

When block_size=4:

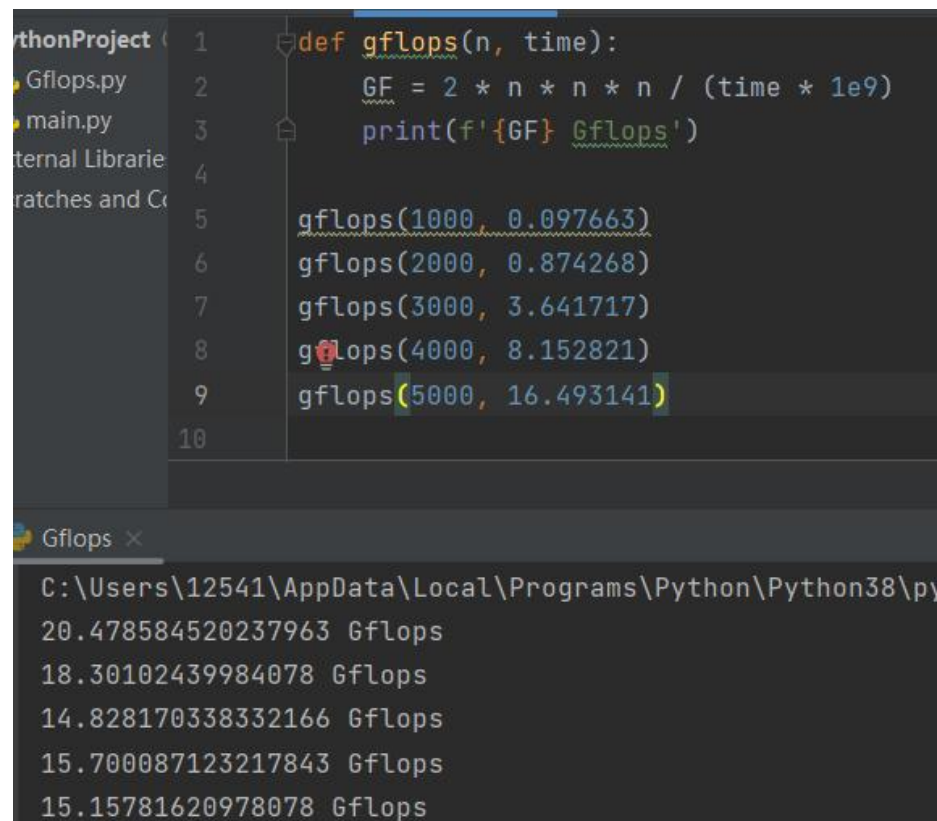
```

[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 1000
n=1000, pad=1
time=0.097663s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 2000
n=2000, pad=1
time=0.874268s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 3000
n=3000, pad=1
time=3.641717s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 4000
n=4000, pad=1
time=8.152821s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 5000
n=5000, pad=1
time=16.493141s

```

[hw020@cluster-001-login-node hw2-hw020]\$./main my_block 1000


```
n=1000, pad=1
time=0.097663s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 2000
n=2000, pad=1
time=0.874268s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 3000
n=3000, pad=1
time=3.641717s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 4000
n=4000, pad=1
time=8.152821s
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 5000
n=5000, pad=1
time=16.493141s
```



The screenshot shows a Python IDE with a file named 'Gflops.py' open. The script defines a function 'gflops(n, time)' that calculates GFlops using the formula $GF = 2 * n * n * n / (time * 1e9)$ and prints the result. The script then calls 'gflops' with five different inputs: (1000, 0.097663), (2000, 0.874268), (3000, 3.641717), (4000, 8.152821), and (5000, 16.493141). Below the code editor, a terminal window titled 'Gflops' shows the output of the script, displaying the calculated GFlops values for each input.

```
def gflops(n, time):
    GF = 2 * n * n * n / (time * 1e9)
    print(f'{GF} Gflops')

gflops(1000, 0.097663)
gflops(2000, 0.874268)
gflops(3000, 3.641717)
gflops(4000, 8.152821)
gflops(5000, 16.493141)
```

```
C:\Users\12541\AppData\Local\Programs\Python\Python38\py
20.478584520237963 Gflops
18.30102439984078 Gflops
14.828170338332166 Gflops
15.700087123217843 Gflops
15.15781620978078 Gflops
```

When block_size=2:

```

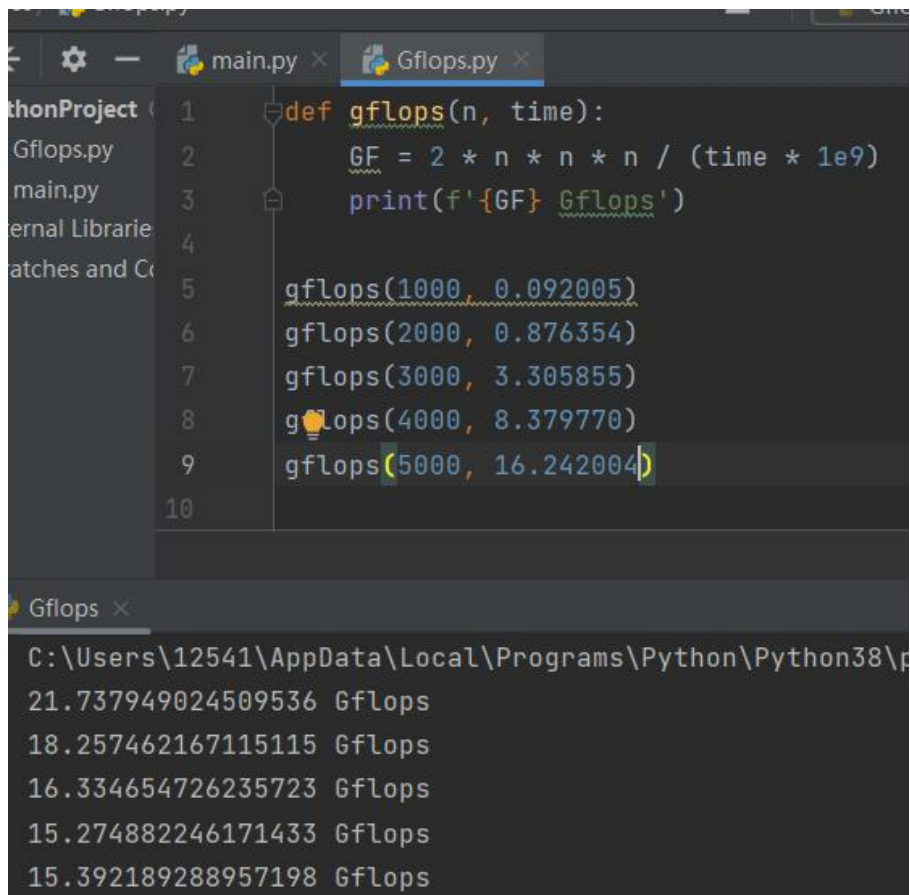
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 1000
n=1000, pad=1
time=0.092005s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 2000
n=2000, pad=1
time=0.876354s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 3000
n=3000, pad=1
time=3.305855s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 4000
n=4000, pad=1
time=8.379770s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 5000
n=5000, pad=1
time=16.242004s

```

```

[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 1000
n=1000, pad=1
time=0.092005s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 2000
n=2000, pad=1
time=0.876354s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 3000
n=3000, pad=1
time=3.305855s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 4000
n=4000, pad=1
time=8.379770s
[hw020@cluster-001-login-node hw2-hw020]$ ./main my_block 5000
n=5000, pad=1
time=16.242004s

```



```
def gflops(n, time):
    GF = 2 * n * n * n / (time * 1e9)
    print(f'{GF} Gflops')

gflops(1000, 0.092005)
gflops(2000, 0.876354)
gflops(3000, 3.305855)
gflops(4000, 8.379770)
gflops(5000, 16.242004)
```

C:\Users\12541\AppData\Local\Programs\Python\Python38\p
21.737949024509536 Gflops
18.257462167115115 Gflops
16.334654726235723 Gflops
15.274882246171433 Gflops
15.392189288957198 Gflops

When block_size=20:

```
Already up to date.  
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 1000  
n=1000, pad=1  
time=0.100313s  
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 2000  
n=2000, pad=1  
time=0.866703s  
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 3000  
n=3000, pad=1  
time=3.280529s  
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 4000  
n=4000, pad=1  
time=8.203220s  
[hwen020@cluster-001-login-node hw2-hwen020]$ ./main my_block 5000  
n=5000, pad=1  
time=16.242142s
```

Seems block_size doesn't work for this.