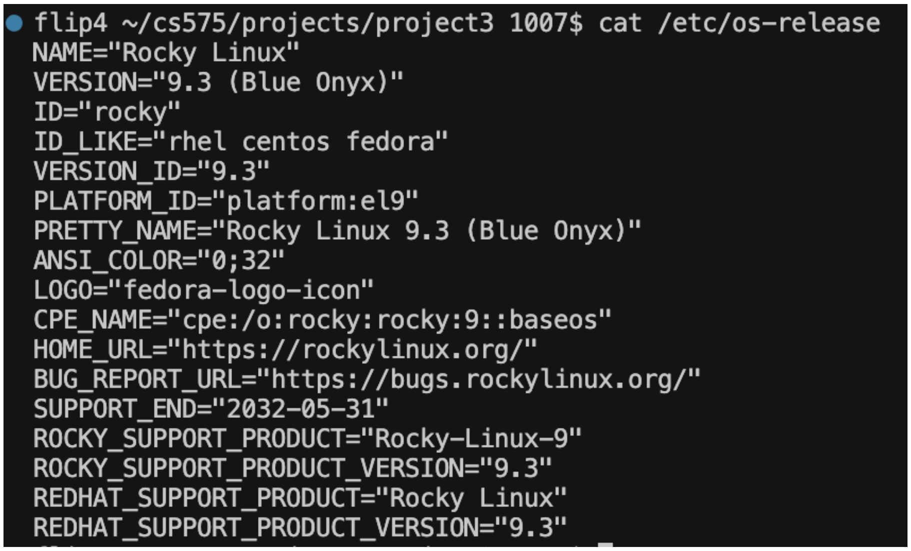
Yen-Chun Chen
chenyenc@oregonstate.edu

Project 03

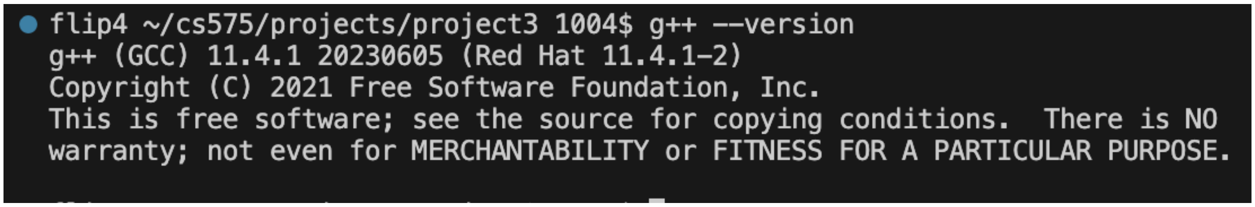
1. Tell what machine you ran this on.
   I ran the code on the COE flip machine.

2. Tell what operating system you were using.



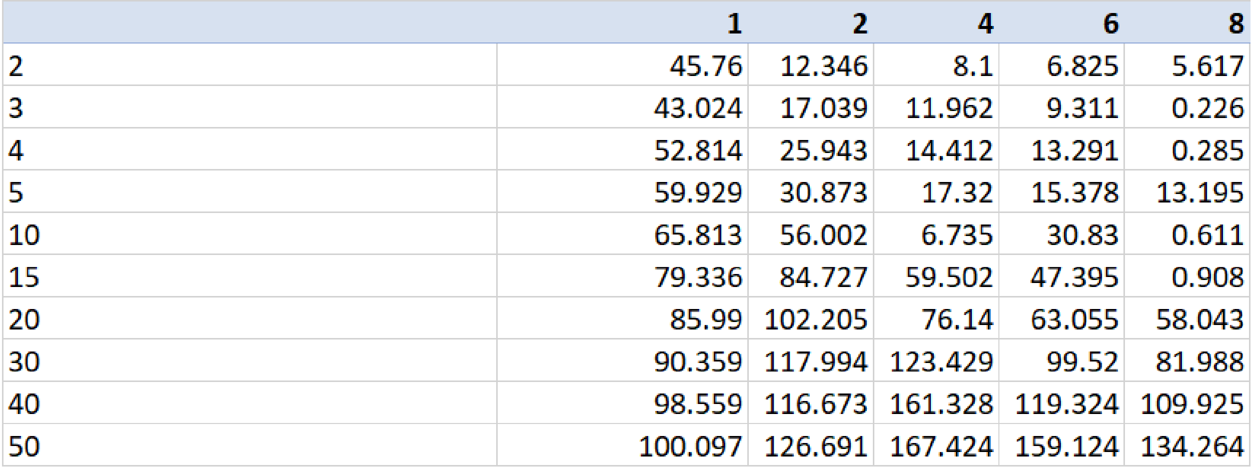
```
flip4 ~/cs575/projects/project3 1007$ cat /etc/os-release
NAME="Rocky Linux"
VERSION="9.3 (Blue Onyx)"
ID="rocky"
ID_LIKE="rhel centos fedora"
VERSION_ID="9.3"
PLATFORM_ID="platform:el9"
PRETTY_NAME="Rocky Linux 9.3 (Blue Onyx)"
ANSI_COLOR="0;32"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:rocky:rocky:9::baseos"
HOME_URL="https://rockylinux.org/"
BUG_REPORT_URL="https://bugs.rockylinux.org/"
SUPPORT_END="2032-05-31"
ROCKY_SUPPORT_PRODUCT="Rocky-Linux-9"
ROCKY_SUPPORT_PRODUCT_VERSION="9.3"
REDHAT_SUPPORT_PRODUCT="Rocky Linux"
REDHAT_SUPPORT_PRODUCT_VERSION="9.3"
```

3. Tell what compiler you used.



```
flip4 ~/cs575/projects/project3 1004$ g++ --version
g++ (GCC) 11.4.1 20230605 (Red Hat 11.4.1-2)
Copyright (C) 2021 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

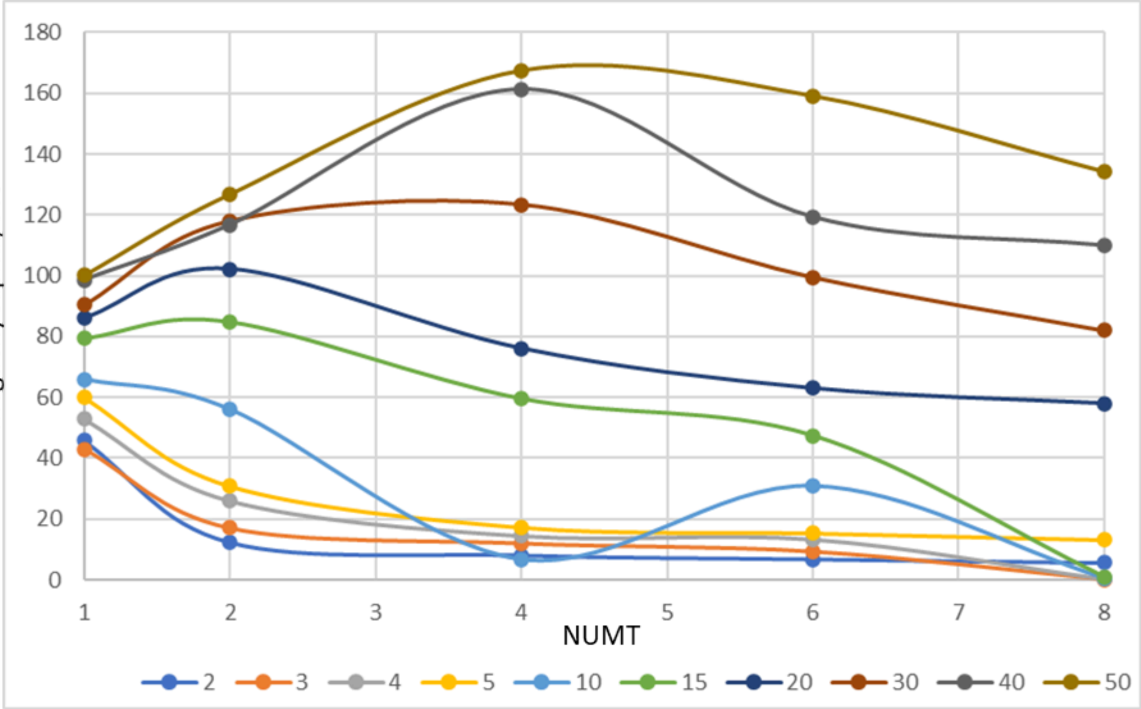
4. Include the table of performance data.



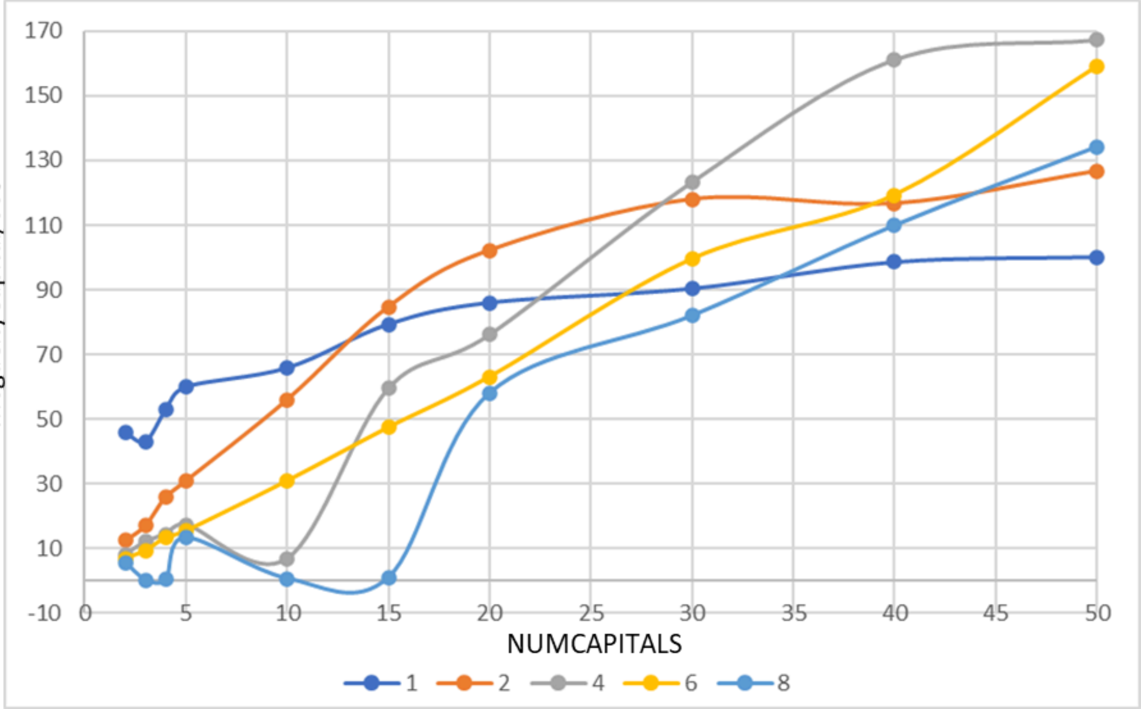
	1	2	4	6	8
2	45.76	12.346	8.1	6.825	5.617
3	43.024	17.039	11.962	9.311	0.226
4	52.814	25.943	14.412	13.291	0.285
5	59.929	30.873	17.32	15.378	13.195
10	65.813	56.002	6.735	30.83	0.611
15	79.336	84.727	59.502	47.395	0.908
20	85.99	102.205	76.14	63.055	58.043
30	90.359	117.994	123.429	99.52	81.988
40	98.559	116.673	161.328	119.324	109.925
50	100.097	126.691	167.424	159.124	134.264

Yen-Chun Chen
chenyenc@oregonstate.edu

5. Include a graph of performance vs. NUMT with the colored curves being NUMCAPITALS.



6. Include a graph of performance vs. NUMCAPITALS cities with the colored curves being NUMT.

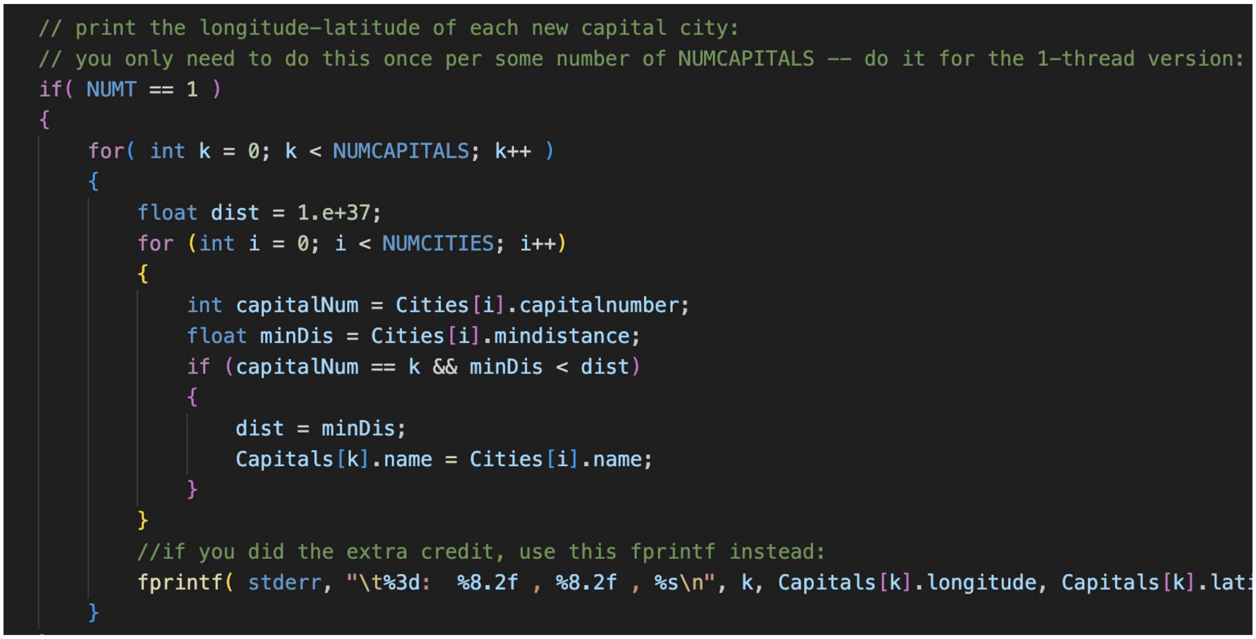


Yen-Chun Chen
chenyenc@oregonstate.edu

7. Tell us what you discovered by doing this. What patterns are you seeing in the graphs?

   The graph does not really show what I expected. I was expecting either 6 or 8 threads will get the best performance since the program has multiple nested loops.
   There could be several reasons such as managing the threads overhead, limited hardware of the flip server, and the K-Means algorithm distributing the data points not evenly among those threads. All things considered, the performance does improve as the number of threads increases but eventually plateaus or decreases due to overhead. Similarly, increasing the number of capitals slows down the performance because of the complexity added by more data points. The optimal performance occurs at a moderate number of threads and capitals, which might be the balance point between parallel efficiency and computation load.

8. Extra Credit to find the closest to each capital's longitude-latitude



```
// print the longitude-latitude of each new capital city:
// you only need to do this once per some number of NUMCAPITALS -- do it for the 1-thread version:
if( NUMT == 1 )
{
    for( int k = 0; k < NUMCAPITALS; k++ )
    {
        float dist = 1.e+37;
        for (int i = 0; i < NUMCITIES; i++)
        {
            int capitalNum = Cities[i].capitalnumber;
            float minDis = Cities[i].mindistance;
            if (capitalNum == k && minDis < dist)
            {
                dist = minDis;
                Capitals[k].name = Cities[i].name;
            }
        }
        //if you did the extra credit, use this fprintf instead:
        fprintf( stderr, "\t%3d:  %8.2f , %8.2f , %s\n", k, Capitals[k].longitude, Capitals[k].lati
    }
```