

**TUGAS PEMROSESAN PARALEL
EKSEKUSI PROGRAM NUMERIK DENGAN MPI PADA UBUNTU
DESKTOP**

Disusun untuk memenuhi tugas Mata Kuliah Pemrosesan Paralel



Disusun Oleh:

Nama : MUHAMMAD ZIHNI ATHALLAH
NIM : 09011282227084

Dosen Pengampu:
Adi Hermansyah, S.Kom., M.T.

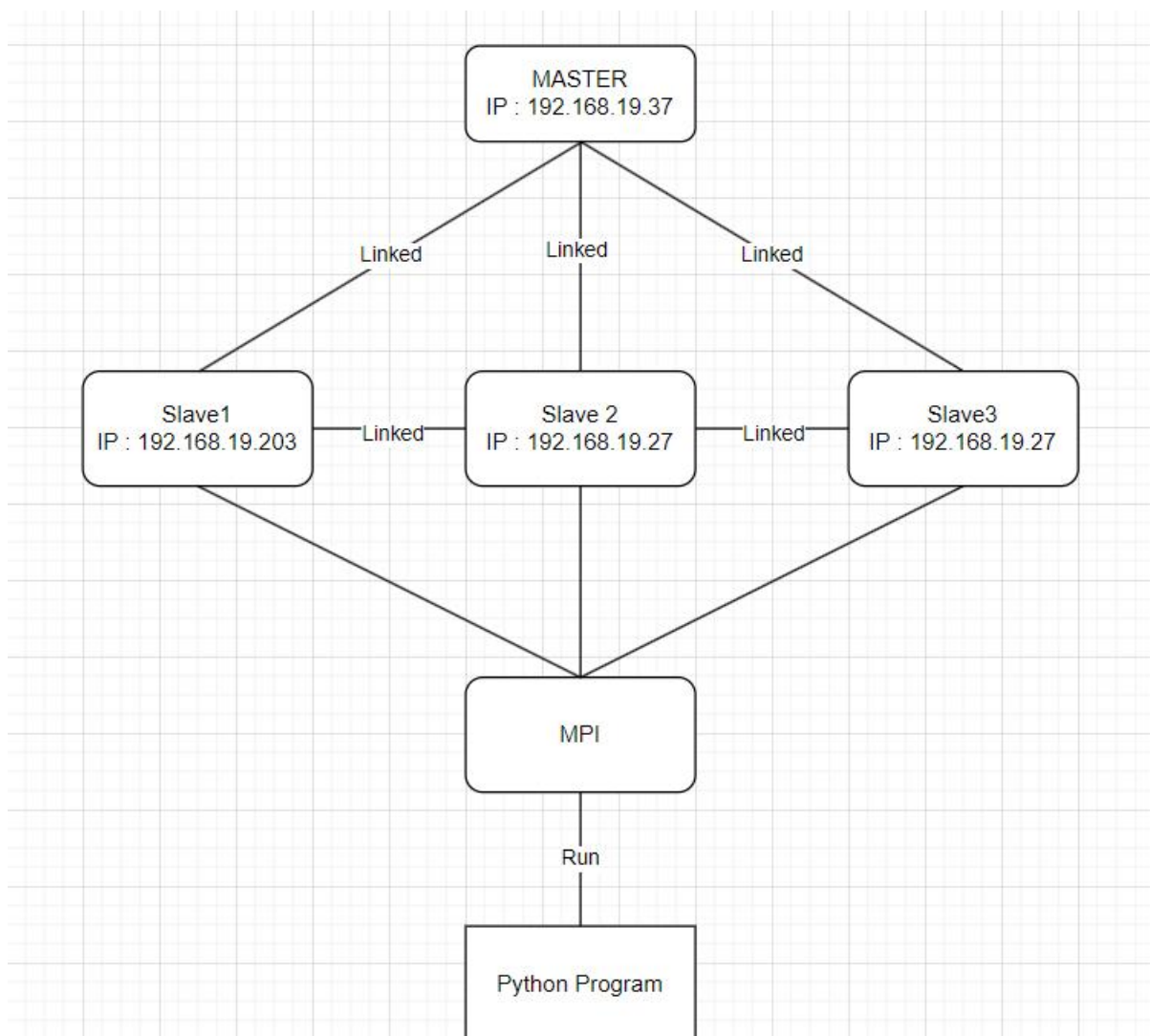
**PROGRAM STUDI SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
PALEMBANG**

2023

Hal yang perlu dipersiapkan

1. Ubuntu Desktop Master
2. 3 Ubuntu Desktop Slave
3. MPI (Master dan Slave)
4. SSH (Master dan Slave)
5. NFS (Master dan Slave)
6. Coding Numerik

Topologi



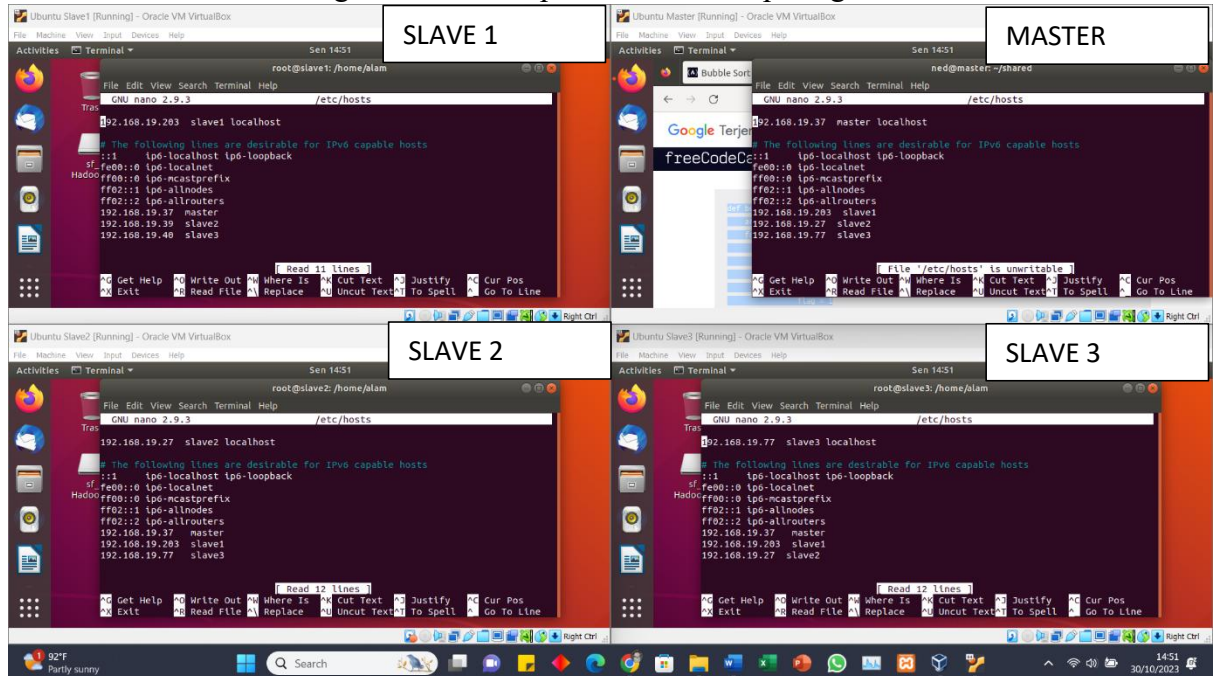
Konfigurasi File etc/hosts

1. Pastikan Ubuntu Desktop yang terinstall menggunakan Network Bridged Adapter, Jika belum kita perlu ganti terlebih dahulu dan setiap Master dan Slave terhubung dengan internet yang sama. Pastikan IP Master dan Slave diketahui

2. Pertama-tama, masukkan command untuk melakukan perubahan pada file /etc/hosts seperti gambar dibawah ini.

```
ned@master:~$ nano /etc/hosts
```

3. Kemudian, lakukan konfigurasi etc/hosts pada etc/hosts, seperti gambar dibawah ini



Membuat User Baru

1. Pada Ubuntu Master dan Slave, perlu dibuat user baru dengan command dibawah ini.

```
ned@master:~$ sudo adduser ned
```

Isi semua permintaan yang diminta sistem, dengan nama user yang harus sama di Master dan setiap Slave

2. Kemudian pada Master dan Slave, masuk ke user yang baru dibentuk dengan perintah berikut.

```
ned@master:~$ su - ned
```

Konfigurasi SSH

1. Pada Ubuntu Master dan Slave, perlu menginstall SSH sendiri terlebih dahulu dengan perintah berikut.

```
ned@master:~$ sudo apt install openssh-server
```

2. Setelah terinstall, dapat dilakukan pengecekan terhadap SSH dengan perintah berikut.

```
ned@master:~$ ssh ned@slave1
```

Sehingga hasilnya akan seperti gambar berikut.

```

ned@master:~$ ssh ned@slave1
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

52 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Fri Nov  3 08:41:02 2023 from 192.168.151.37
ned@slave1:~$

```

Generate & Copy Keygen from Master to Slave

1. Pada Ubuntu Master, lakukan perintah berikut

```
ned@master:~$ sudo apt install openssh-server
```

Nantinya sistem akan meminta beberapa input, lewati saja. Nantinya, akan ada folder .ssh berisi file id_rsa dan id_rsa.pub

2. Kemudian pada Ubuntu Master, copy isi file id_rsa.pub ke file authorized_keys dengan ssh seperti gambar berikut

```
ned@master:~$ cd .ssh
```

```
ned@master:~$ cat id_rsa.pub | ssh ned@master "mkdir .ssh; cat >> .ssh/authorized_keys"
```

Lakukan copy paste berkali-kali dari Master ke Slave dengan mengubah-ubah host tersebut.

Konfigurasi NFS

1. Pada Ubuntu Master dan Slave, perlu membuat suatu direktori dengan perintah berikut.

```
ned@master:~$ mkdir shared
```

2. Kemudian pada Ubuntu Master, lakukan instalasi NFS Server dengan perintah berikut.

```
ned@master:~$ sudo apt install nfs-kernel-server
```

3. Masih pada Ubuntu Master, buka file /etc/exports dengan nano, dan tambahkan baris seperti dibawah ini.

```
ned@master:~$ nano /etc/exports
```



```
ned@master: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 /etc/exports  
# /etc/exports: the access control list for filesystems which may be exported  
# to NFS clients. See exports(5).  
#  
# Example for NFSv2 and NFSv3:  
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_che$  
#  
# Example for NFSv4:  
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)  
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)  
#  
/home/ned/shared *(rw,sync,no_root_squash,no_subtree_check)
```

4. Setelah melakukan konfigurasi pada file /etc/exports, masukkan perintah berikut.

```
ned@master:~$ sudo exportfs -a  
ned@master:~$ sudo systemctl restart nfs-kernel-server
```

5. Kemudian pada Ubuntu Slave, lakukan penginstalan NFS Client dengan perintah berikut.

```
ned@master:~$ sudo apt install nfs-common
```

6. Terakhir, lakukan Mounting pada 3 Ubuntu Slave dengan perintah berikut.

```
ned@master:~$ sudo mount master:/home/ned/shared/ /home/ned/shared/
```

Eksekusi Program Numerik

1. Pada Ubuntu Master dan Server, lakukan perintah berikut untuk instalasi MPI.

```
ned@master:~$ sudo apt install openssh-server
```

2. Kemudian pada Ubuntu Master, buat suatu file python pada folder shared dengan perintah dibawah ini.

```
ned@master:~$ cd shared/  
ned@master:~/shared$ sudo gedit sort.py
```

3. Pastikan sudah melakukan instalasi numpy untuk mengeksekusi program numerik

```
ned@master:~$ pip install numpy  
Collecting numpy  
Using cached https://files.pythonhosted.org/packages/3a/5f/47e578b3ae79e2624e205445ab77a1848acdaa2929a00eeef6b16eaaeb20/numpy-1.16.6-cp27-cp27mu-manylinux1_x86_64.whl  
Installing collected packages: numpy  
Successfully installed numpy-1.16.6
```

4. Sebelum mengeksekusi codingan numerik, terlebih dahulu melakukan testing pada MPI dengan perintah berikut

```
ned@master:~/shared$ mpiexec -oversubscribe -host master,slave1,slave2,slave3 python -m mpi4py.bench helloworld  
Hello, World! I am process 0 of 4 on master.  
Hello, World! I am process 1 of 4 on slave1.  
Hello, World! I am process 2 of 4 on slave2.  
Hello, World! I am process 3 of 4 on slave3.
```

5. Masukkan codingan numerik pada file python tersebut dengan perintah dibawah ini.

```
from mpi4py import MPI
import numpy as np

def matrix_multiply(A, B):
    return np.dot(A, B)

if __name__ == '__main__':
    comm = MPI.COMM_WORLD
    rank = comm.Get_rank()

    if rank == 0:
        # Inisialisasi matriks A dan B dengan angka integer acak antara 0 dan 100
        rows, cols = 3, 3
        A = np.random.randint(0, 11, size=(rows, cols))
        B = np.random.randint(0, 11, size=(cols, rows))
    else:
        A, B = None, None

    A = comm.bcast(A, root=0)
    B = comm.bcast(B, root=0)

    local_result = matrix_multiply(A, B)

    if rank == 0:
        global_result = np.zeros_like(local_result)
    else:
        global_result = None

    comm.Reduce(local_result, global_result, op=MPI.SUM, root=0)

    if rank == 0:
        print("Matriks A:")
        print(A)
        print("Matriks B:")
        print(B)
        print("Hasil perkalian matriks:")
        print(global_result)
```

6. Selanjutnya, jalankan file tersebut dengan perintah berikut sehingga muncul output berikut.

- Eksekusi tanpa MPI

```
ned@master:~/shared$ python sort.py
Matriks A:
[[1 3 9 ... 5 7 9]
 [4 8 7 ... 3 6 1]
 [8 5 2 ... 4 6 8]
 ...
 [8 2 4 ... 7 0 2]
 [2 9 2 ... 7 4 3]
 [5 8 0 ... 7 7 9]]
Matriks B:
[[6 6 1 ... 1 9 5]
 [1 0 0 ... 6 0 1]
 [1 9 9 ... 6 9 6]
 ...
 [7 6 3 ... 5 4 6]
 [1 3 5 ... 1 5 3]
 [9 7 4 ... 4 3 9]]
Hasil perkalian matriks:
[[1061 1129 974 ... 1097 1167 1305]
 [1081 1336 1205 ... 1149 1327 1354]
 [ 925 1166 1055 ... 977 1026 1227]
 ...
 [1178 1304 1224 ... 1203 1321 1316]
 [1000 1297 1162 ... 1084 1190 1158]
 [1182 1299 1182 ... 1251 1225 1268]]
Waktu Eksekusi : 0.000366 detik
```

- Eksekusi dengan MPI Master Only

```

ned@master:~/shared$ mpiexec -np 1 -host master python sort.py
Matriks A:
[[5 1 5 ... 0 0 9]
 [2 2 0 ... 9 0 2]
 [6 0 3 ... 8 1 0]
 ...
 [7 8 7 ... 0 9 5]
 [6 9 0 ... 2 0 9]
 [2 8 7 ... 3 5 8]]
Matriks B:
[[4 9 1 ... 3 0 9]
 [8 9 4 ... 5 8 7]
 [3 7 2 ... 2 3 0]
 ...
 [7 8 4 ... 9 0 2]
 [8 6 1 ... 3 4 8]
 [9 0 5 ... 8 2 9]]
Hasil perkalian matriks:
[[1248 1358 1358 ... 1387 1249 1393]
 [1032 1062 1066 ... 1158 963 1117]
 [1403 1428 1224 ... 1466 1161 1407]
 ...
 [1460 1545 1335 ... 1520 1349 1581]
 [1342 1395 1328 ... 1582 1350 1619]
 [1516 1528 1457 ... 1581 1353 1531]]
Waktu Eksekusi : 0.000400 detik

```

- Eksekusi dengan MPI Master & Slave1

```

ned@master:~/shared$ mpiexec -np 2 -host master,slave1 python sort.py
Matriks A:
[[8 4 4 ... 4 5 4]
 [7 7 7 ... 2 1 9]
 [0 7 0 ... 7 0 6]
 ...
 [2 0 0 ... 4 6 0]
 [7 8 6 ... 8 1 0]
 [5 0 5 ... 0 4 9]]
Matriks B:
[[1 4 1 ... 5 4 2]
 [0 5 6 ... 5 5 6]
 [5 6 1 ... 5 5 1]
 ...
 [8 8 2 ... 4 4 2]
 [8 2 7 ... 3 7 2]
 [5 4 1 ... 2 6 7]]
Hasil perkalian matriks:
[[2194 2301 2264 ... 2424 2463 2428]
 [2298 2591 2398 ... 2572 2549 2581]
 [2425 2594 2462 ... 2898 3028 2712]
 ...
 [2464 3004 2431 ... 2895 3061 2727]
 [2227 2874 2378 ... 2743 2807 2544]
 [2466 2776 2466 ... 2922 3056 2753]]
Waktu Eksekusi : 0.014325 detik

```

- Eksekusi dengan MPI Master, Slave1, & Slave2

```

ned@master:~/shared$ mpiexec -np 3 -host master,slave1,slave2 python sort.py
Matriks A:
[[7 6 5 ... 3 7 4]
 [0 3 5 ... 3 5 4]
 [7 8 2 ... 7 9 0]
 ...
 [9 3 4 ... 5 5 0]
 [9 2 5 ... 0 5 1]
 [6 7 2 ... 3 5 4]]
Matriks B:
[[7 2 6 ... 3 4 8]
 [5 6 8 ... 2 8 1]
 [0 9 0 ... 7 0 6]
 ...
 [1 7 2 ... 4 3 5]
 [8 9 6 ... 2 5 6]
 [5 4 4 ... 1 5 5]]
Hasil perkalian matriks:
[[3596 3517 3881 ... 3571 3905 3237]
 [3703 3456 3896 ... 3992 3955 3859]
 [3655 3717 3836 ... 3563 3987 3448]
 ...
 [3831 3780 3995 ... 4044 3946 3644]
 [3794 3792 3821 ... 3560 3643 3442]
 [3572 3936 4016 ... 3841 3924 3623]]
Waktu Eksekusi : 0.014313 detik

```

- Eksekusi dengan MPI Master, Slave1, Slave2, & Slave3

```
ned@master:~/shared$ mpiexec -np 4 -host master,slave1,slave2,slave3 python sort
.py
Matriks A:
[[8 7 0 ... 3 3 9]
 [3 5 0 ... 4 6 1]
 [7 9 3 ... 9 7 4]
 ...
 [0 8 1 ... 5 1 9]
 [6 1 7 ... 0 0 2]
 [2 9 9 ... 4 2 3]]
Matriks B:
[[4 6 1 ... 7 8 8]
 [5 1 5 ... 5 5 4]
 [6 0 8 ... 5 3 4]
 ...
 [2 4 8 ... 7 8 2]
 [5 0 2 ... 6 1 7]
 [4 8 6 ... 8 3 9]]
Hasil perkalian matriks:
[[5779 5181 5576 ... 5344 5124 5435]
 [5126 4716 5265 ... 4842 5107 5240]
 [5682 5268 5609 ... 5315 5519 5705]
 ...
 [5184 5201 5545 ... 5090 5213 5501]
 [4799 5072 5139 ... 4746 5095 5188]
 [5191 5149 4964 ... 4743 5068 5364]]
Waktu Eksekusi : 0.023382 detik
```