



# Klasifikasi multiprosesor

Sistem multiprosesor dapat dibedakan menjadi dua kelompok:

1. Loosely coupled: memori tidak disharing dan setiap prosesor mempunyai memori sendiri.
2. Tightly coupled: melakukan sharing informasi melalui sebuah memori bersama. Klasifikasi tightly coupled terbagi menjadi :
  - Uniform memory acces (UMA) : waktu akses memori sama untuk semua prosesor
  - Non uniform memory acces (NUMA) : waktu akses memori berbeda untuk prosesor yang berbeda.
  - No Remote Memory Access (NORMA)

# CLUSTER

- Suatu interkoneksi sistem komputer multi independen yang dioperasikan seperti sebuah sistem tunggal dalam suatu kerjasama. Setiap simpul dalam cluster dapat juga bekerja secara independen
- Cluster berbeda dengan sebuah jaringan komputer. Pada jaringan komputer tujuan utama adalah resource sharing sementara cluster menyediakan semua kebaikan dari sistem mutiprosessor

# CLUSTER

Cluster komputer terbagi ke dalam beberapa kategori, sebagai berikut:

## 1. *High-availability clusters*

- Diimplementasikan untuk tujuan meningkatkan ketersediaan layanan yang disediakan oleh cluster itu sendiri. Ukuran yang umum digunakan adalah dua node.

## 2. *Load-balancing clusters*

- Mendistribusikan beban kerja secara merata melalui beberapa node yang bekerja dibelakang.

## 3. *Computer clusters*

- Tujuan clustering adalah untuk tujuan komputasi tidak mengedepankan layanan basis data/web.

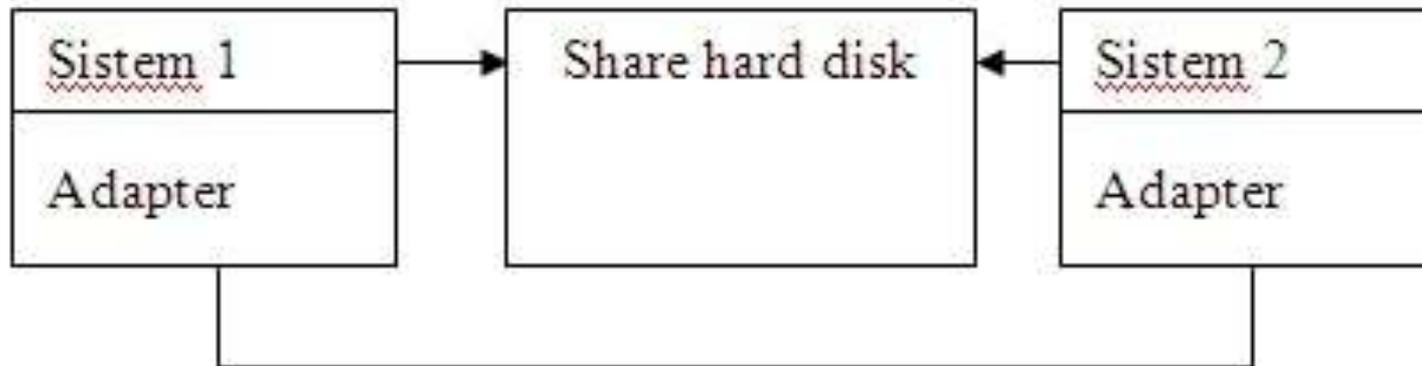
## 4. *Grid computing*

- Mencakup banyak pekerjaan independen yang tidak harus berbagi data yang sama. Berbagi sumber daya mungkin dapat diterapkan namun hasil dari setiap proses tidak mempengaruhi kegiatan yang lainnya.

# CLUSTER

Ada dua tujuan utama dibelakang terbentuknya cluster:

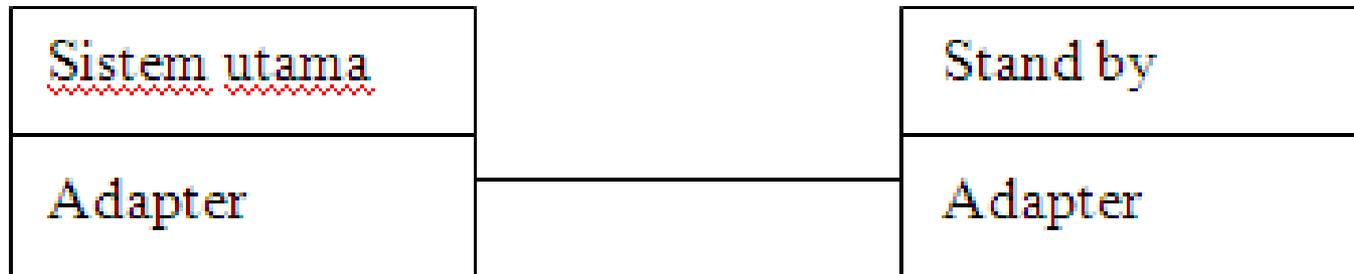
1. Load sharing: dua sistem membentuk cluster dan sharing beban pemrosesan.



Gambar konsep load sharing pada cluster

# CLUSTER

2. Fault Tolerance: dua sistem yang membentuk cluster dimana salah satu digunakan sebagai hot stand-by yang berfungsi melakukan pengambil alihan fungsi bila terjadi kegagalan. Dan berlaku diam selama tidak menemukan kegagalan pada sistem lain.

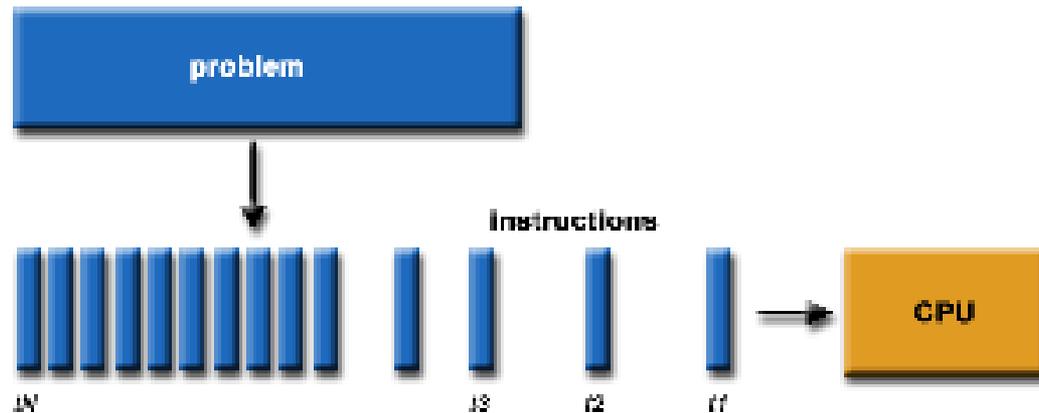


Gambar konsep fault tolerance

# Komputasi Paralel

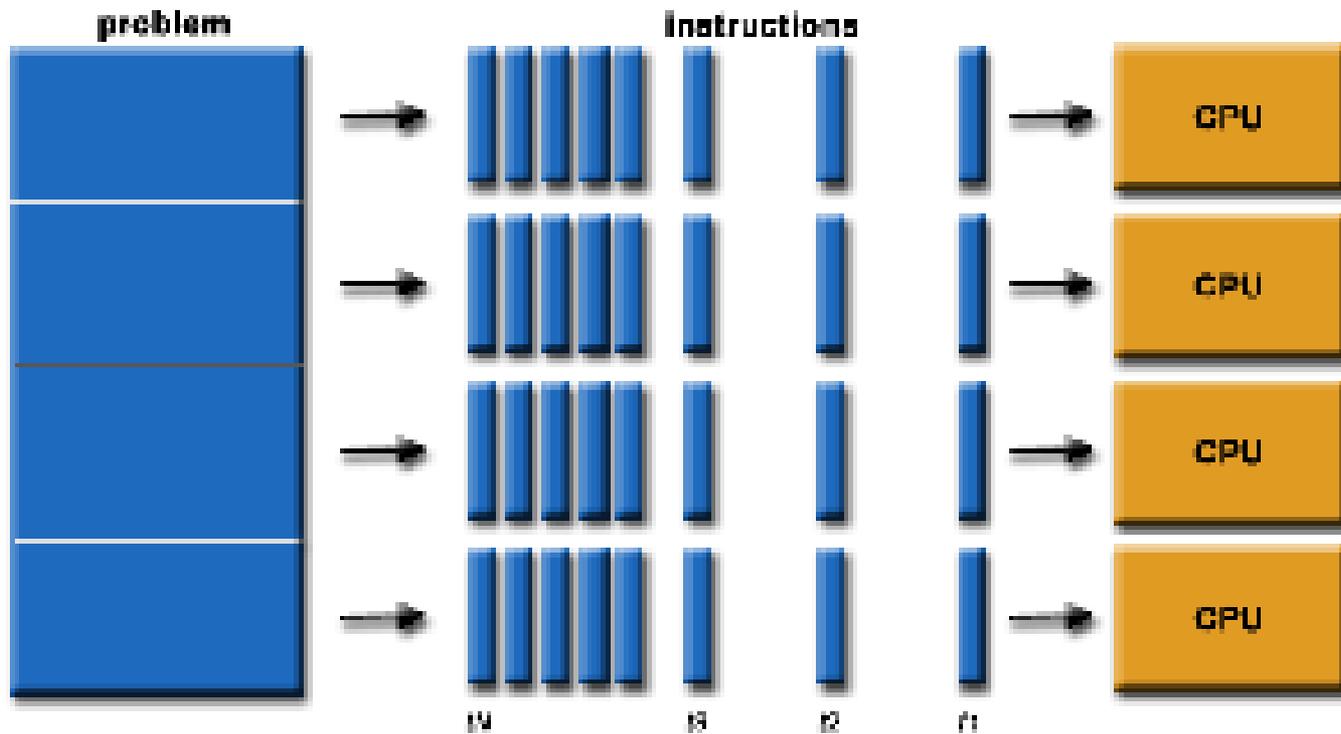
**Komputasi paralel** adalah salah satu teknik melakukan komputasi secara bersamaan dengan memanfaatkan beberapa komputer independen secara bersamaan. Perbedaan antara komputasi tunggal dengan komputasi paralel, bisa digambarkan pada gambar di bawah ini:

Gambar. Penyelesaian Sebuah Masalah pada Komputasi Tunggal



# Komputasi Paralel

Gambar Penyelesaian Sebuah Masalah pada Komputasi Paralel



# Komputasi Paralel

- ❖ Pengolahan Paralel : pengolahan informasi yang menekankan pada manipulasi data-data elemen secara simultan.
- ❖ Tujuan : untuk mempercepat komputasi dari sistem komputer dan menambah jumlah keluaran ( throughput ) yang dapat dihasilkan dalam jangka waktu tertentu.
- ❖ Komputer Paralel : Komputer yang memiliki kemampuan untuk melakukan pengolahan paralel.

# Komputasi Paralel

Peningkatan throughput dapat dilakukan dengan:

- Meningkatkan kecepatan operasi
- Meningkatkan jumlah operasi yang dapat dilakukan dalam satu waktu tertentu (concurrency).

# Klasifikasi Komputer Paralel

**Klasifikasi Komputer parallel dalam beberapa sudut pandang:**

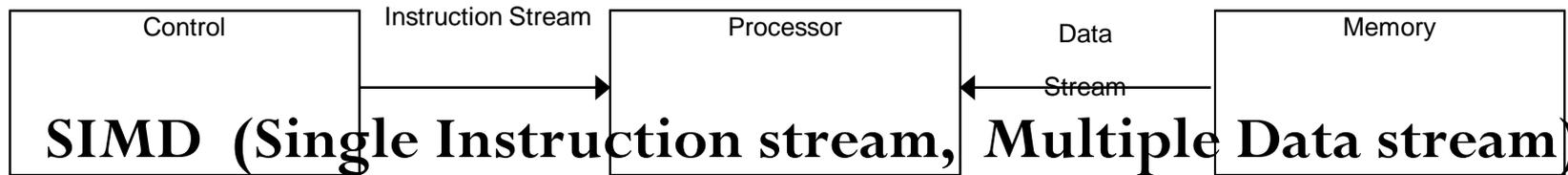
## ➤ **Klasifikasi Flynn**

Memandang organisasi sebuah sistem komputer berdasarkan jumlah instruksi dan data ang dimanipulasi secara simultan dan membaginya menjadi empat kelompok utama:

# Klasifikasi Komputer Paralel

## 1. SISD (Single Instruction stream, Single Data stream)

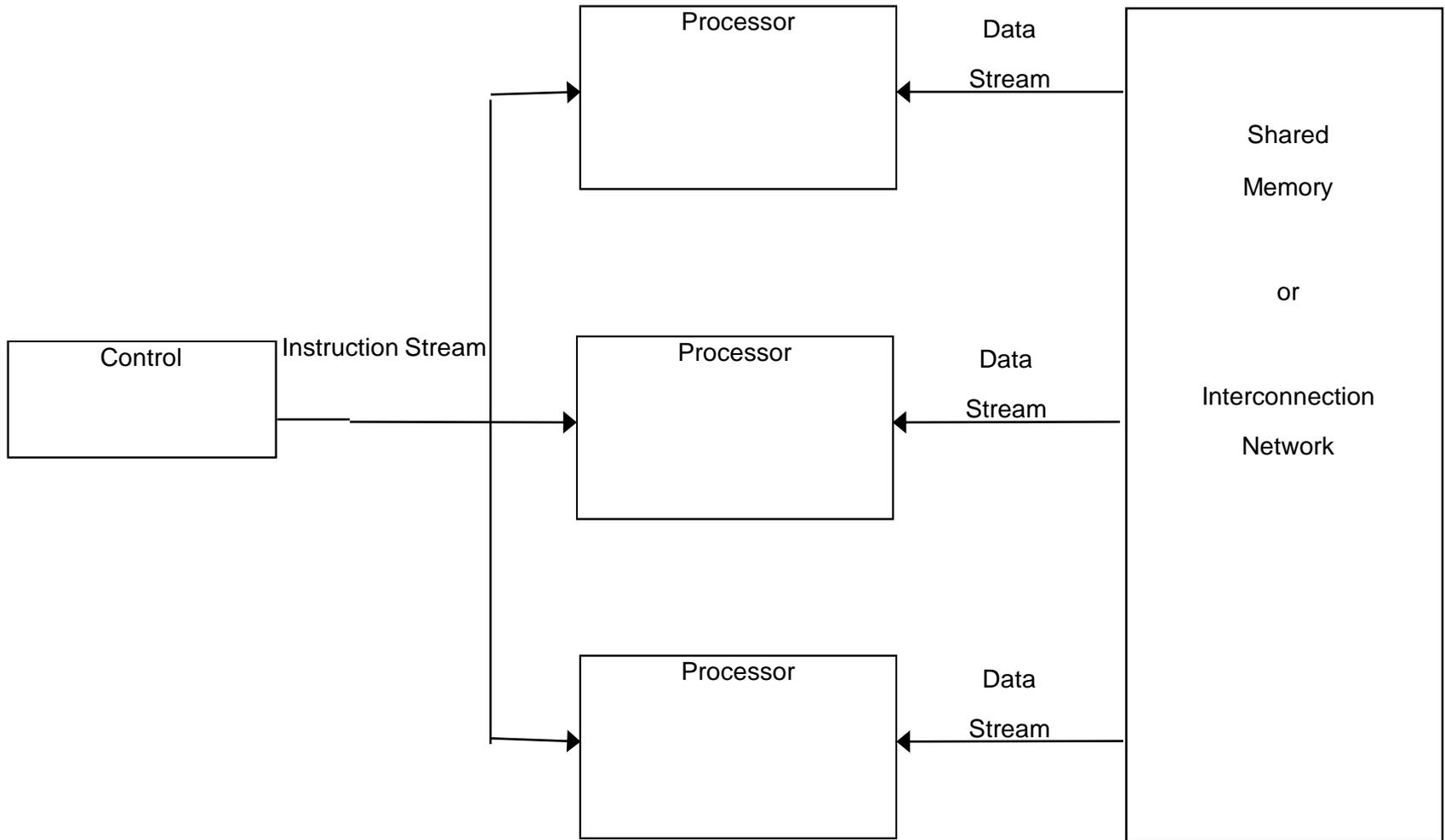
- Komputer tunggal yang mempunyai satu unit kontrol, satu unit prosesor dan satu unit memori.



## 2. SIMD (Single Instruction stream, Multiple Data stream)

- Komputer yang mempunyai beberapa unit prosesor di bawah pengawasan satu unit kontrol. Setiap prosesor menerima instruksi yang sama dari unit kontrol, tetapi beroperasi pada data yang berbeda.

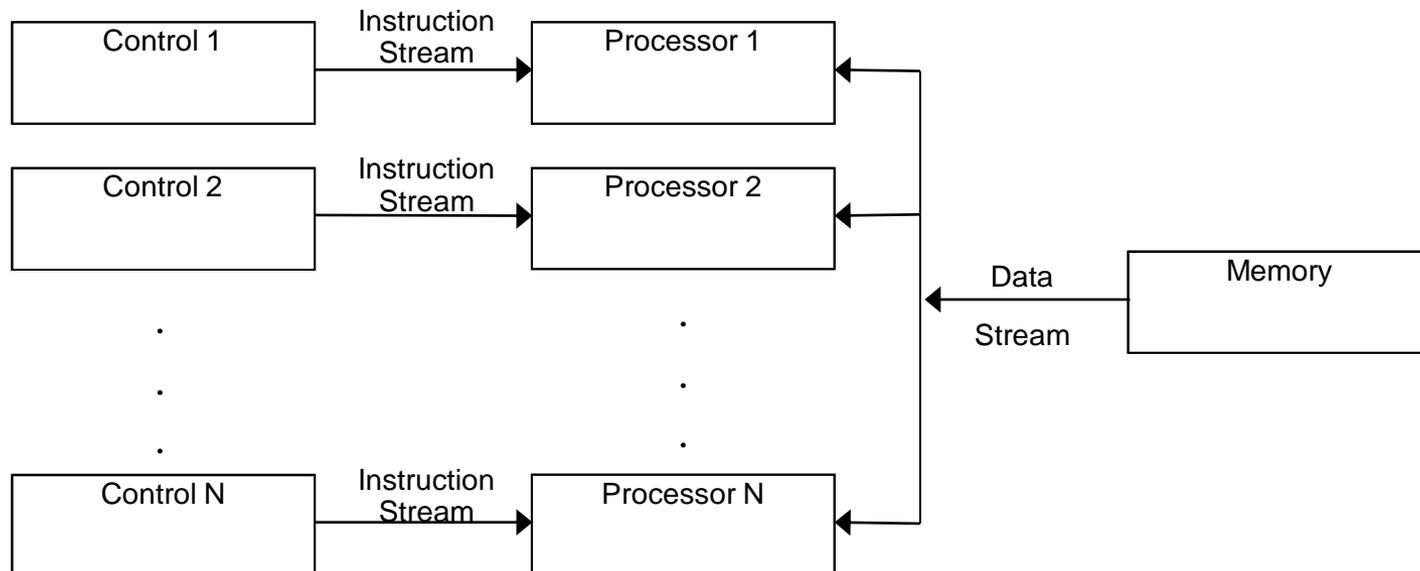
# Klasifikasi Komputer Paralel



# Klasifikasi Komputer Paralel

## 3. MISD (Multiple Instruction stream, Single Data stream).

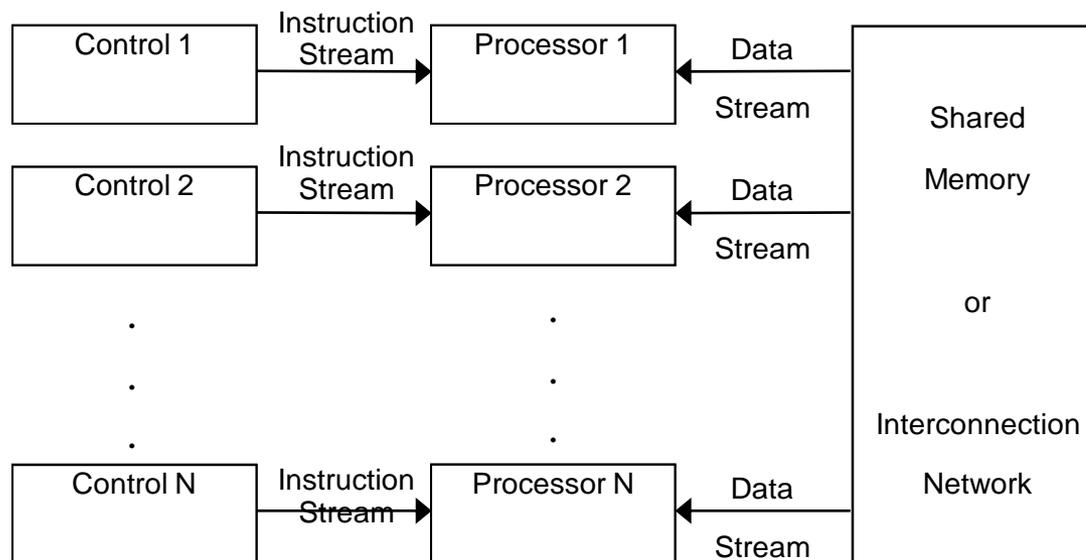
- Sampai saat ini struktur ini masih merupakan struktur teoritis dan belum ada komputer dengan model ini.



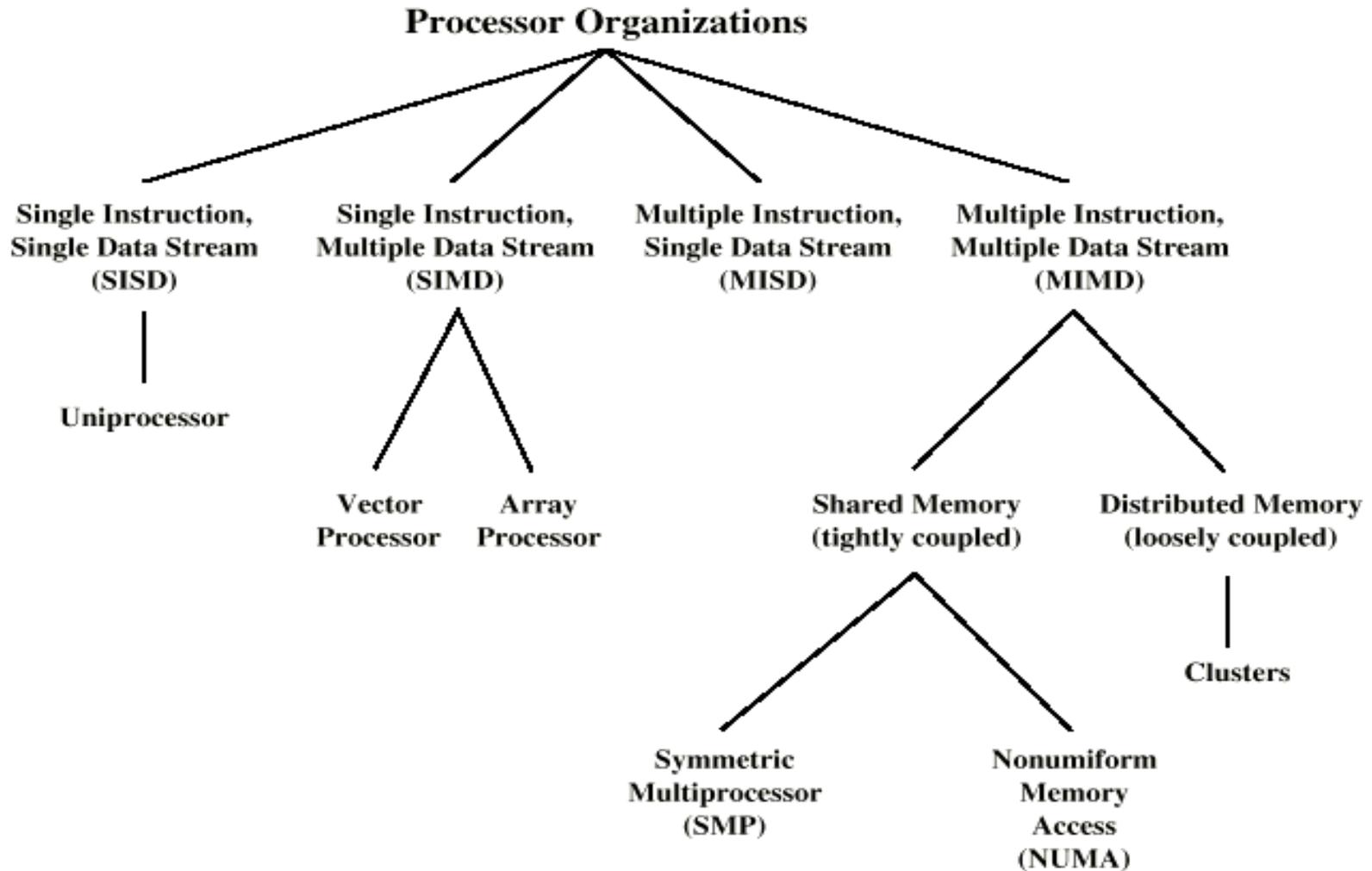
# Klasifikasi Komputer Paralel

## 4. MIMD (*Multiple Instruction stream, Multiple Data stream*)

- Organisasi komputer yang memiliki kemampuan untuk memproses beberapa program dalam waktu yang sama. Pada umumnya multiprosesor dan multikomputer termasuk dalam kategori ini.



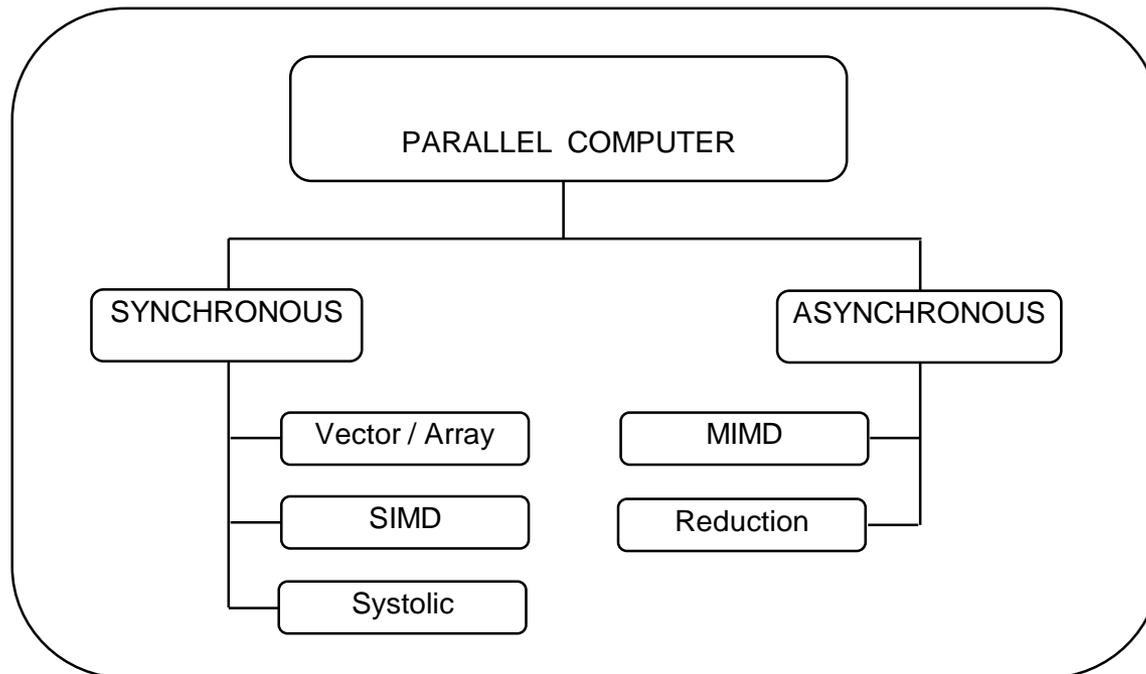
# Klasifikasi Komputer Paralel



# Klasifikasi Komputer Paralel

## ➤ T.G. LEWIS

- T.G. Lewis membedakan komputer paralel ke dalam dua kelas, berdasarkan ada atau tidak adanya common global clock, sebagai : **Synchronous dan Asynchronous.**



# Klasifikasi Komputer Paralel

## **Synchronous :**

- Pada komputer paralel yang termasuk dalam kategori ini terdapat koordinasi yang mengatur beberapa operasi untuk dapat berjalan bersamaan sedemikian hingga tidak ada ketergantungan antar operasi.
- Parallelism yang termasuk dalam kategori ini adalah vector/array parallelism, SIMD dan systolic parallelism.
- Systolic parallel computer adalah multiprocessor dimana data didistribusikan dan dipompa dari memory ke suatu array prosesor sebelum kembali ke memori.

# Klasifikasi Komputer Paralel

## **Asynchronous :**

- Pada komputer paralel yang termasuk dalam kategori asynchronous, masing-masing prosesor dapat diberi tugas atau menjalankan operasi berbeda dan masing-masing prosesor melaksanakan operasi tersebut secara sendiri-sendiri tanpa perlu koordinasi.
- Paradigma yang juga termasuk dalam kategori ini adalah MIMD dan reduksi.
- Paradigma reduksi adalah paradigma yang berpijak pada konsep graph reduksi. Program dengan model reduksi diekspresikan sebagai graph alur data. Komputasi berlangsung dengan cara mereduksi graph dan program berhenti jika graph akhirnya hanya mempunyai satu simpul.

# Klasifikasi Komputer Paralel

## ➤ MICHAEL J. QUINN

Quinn membedakan paralelisma ke dalam dua jenis : Data Parallelism dan Control Parallelism.

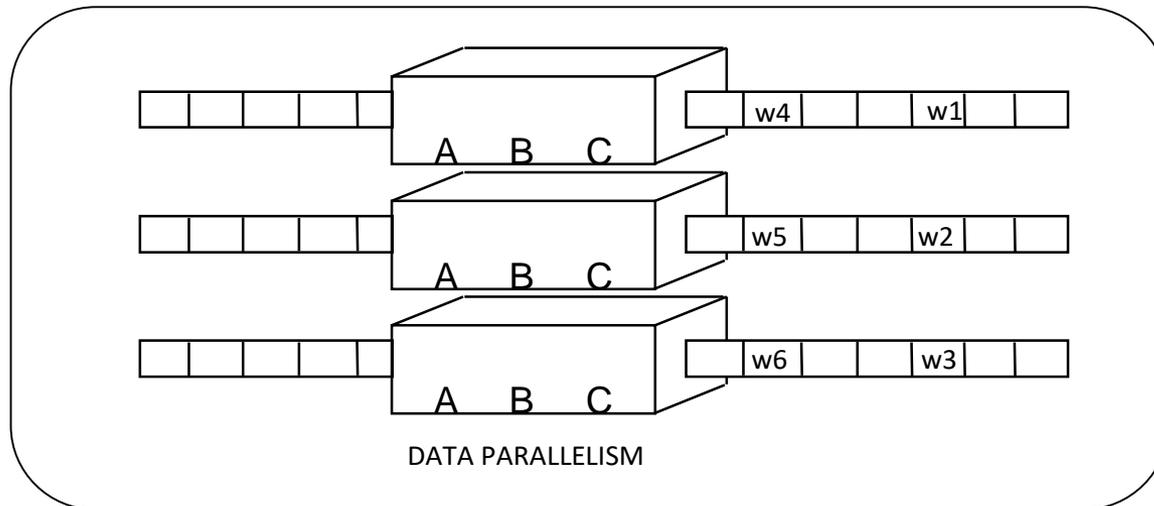
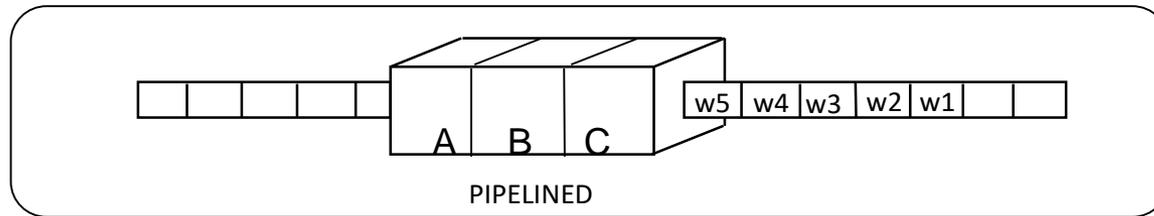
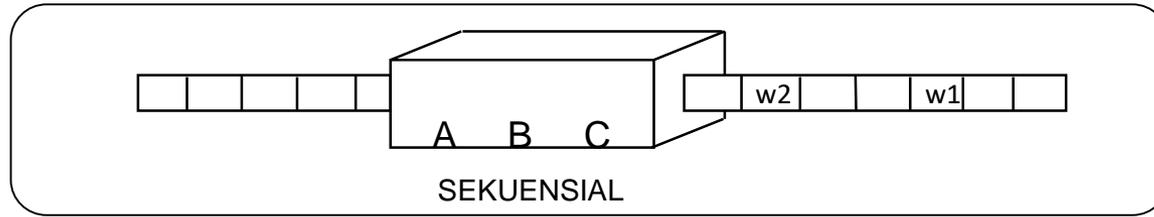
1. **Data Parallelism** : penerapan operasi yang sama secara simultan terhadap elemen-elemen dari kumpulan data.
2. **Control Parallelism** : penerapan operasi-operasi berbeda terhadap elemen-elemen data yang berbeda secara bersamaan. Pada control parallelism dapat terjadi aliran data antar proses-proses dan kemungkinan terjadi aliran data yang kompleks/rumit.

# Klasifikasi Komputer Paralel

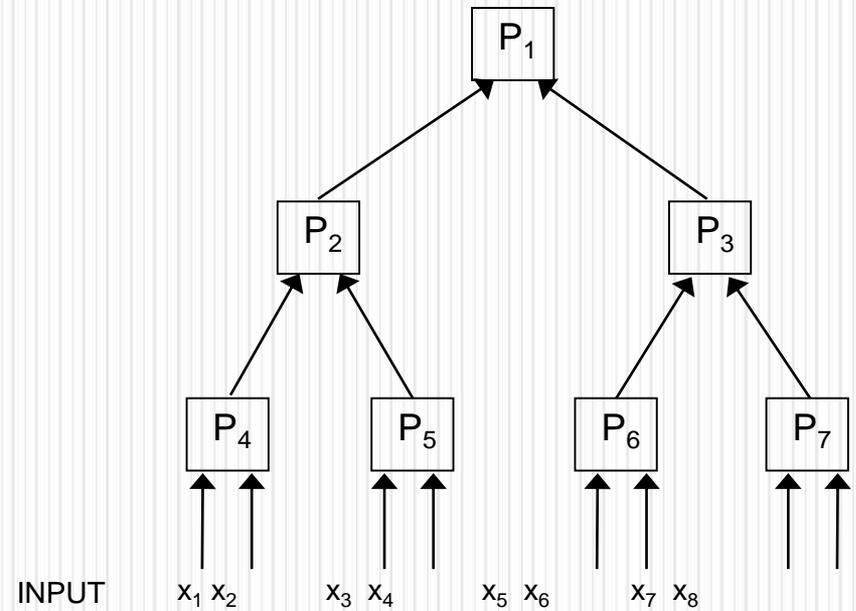
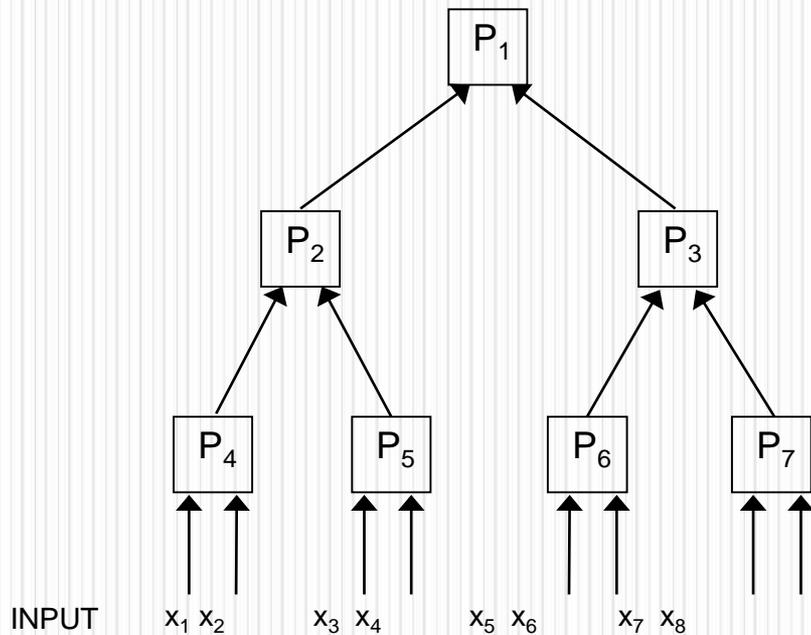
Teknik pemecahan satu tugas menjadi beberapa sub-tugas dan mengeksekusi tugas tersebut secara bersamaan dalam unit-unit multihardware atau segmen-segmen yang disebut sebagai Pipelining merupakan satu kasus khusus dari control parallelism dimana aliran data membentuk jalur yang sederhana.

# Klasifikasi Komputer Paralel

Gambar Ilustrasi perbandingan pipelining dengan data parallelism



# ANALISA ALGORITMA PARALEL



# Analisa Algoritma Paralel

## **Analisa Algoritma Paralel.**

Pada saat sebuah algoritma digunakan untuk memecahkan sebuah problem, maka performance dari algoritma tersebut akan dinilai. Hal ini berlaku untuk algoritma sekuensial maupun algoritma paralel. Penampilan sebuah algoritma pengolahan paralel dapat dinilai dari beberapa kriteria, seperti running time dan banyaknya prosesor yang digunakan.

# Analisa Algoritma Paralel

## ➤ **Running Time**

Running time adalah waktu yang digunakan oleh sebuah algoritma untuk menyelesaikan masalah pada sebuah komputer paralel dihitung mulai dari saat algoritma mulai hingga saat algoritma berhenti. Jika prosesor-prosesornya tidak mulai dan selesai pada saat yang bersamaan, maka running time dihitung mulai saat komputasi pada prosesor pertama dimulai hingga pada saat komputasi pada prosesor terakhir selesai.

# Analisa Algoritma Paralel

## Counting Steps

Untuk menentukan running time, secara teoritis dilakukan analisa untuk menentukan waktu yang dibutuhkan sebuah algoritma dalam mencari solusi dari sebuah masalah. Hal ini dilakukan dengan cara menghitung banyaknya operasi dasar, atau step, yang dilakukan oleh algoritma untuk keadaan terburuknya (worst case).

# Analisa Algoritma Paralel

Langkah-langkah yang diambil oleh sebuah algoritma dibedakan ke dalam dua jenis yaitu :

1. Computational step

Sebuah computational step adalah sebuah operasi aritmetika atau operasi logika yang dilakukan terhadap sebuah data dalam sebuah prosesor.

2. Routing step.

Pada routing step, sebuah data akan melakukan perjalanan dari satu prosesor ke prosesor lain melalui shared memory atau melalui jaringan komunikasi.

# Analisa Algoritma Paralel

Contoh 1 :

Perhatikan sebuah file komputer dengan  $n$  entri berbeda.

Pada file tersebut akan diperiksa apakah  $x$  terdapat di dalamnya. Dengan algoritma sekuensial, keadaan terburuknya (worst case) untuk menemukan  $x$  membutuhkan  $n$  langkah, dimana tiap langkah adalah membandingkan  $x$  dengan sebuah entri pada file.

# Analisa Algoritma Paralel

Keadaan terburuk terjadi jika  $x$  ternyata sama dengan entri terakhir pada file atau  $x$  tidak terdapat pada file tersebut.

Dengan EREW SM SIMD (Exclusive Read Exclusive Write Shared Memory SIMD) komputer dengan  $N$  prosesor, dimana  $N \leq n$ , pada worst casenya dibutuhkan  $\log N + n/N$  langkah.

Misalkan  $P_1, P_2, \dots, P_N$  prosesor-prosesor pada EREW SM SIMD komputer tersebut.

# Analisa Algoritma Paralel

Proses pencarian entri yang sama dengan  $x$  adalah :

➤ Broadcasting,  $x$  dikomunikasikan pada semua prosesor dengan cara :

1.  $P_1$  membaca  $x$  dan mengkomunikasikan dengan  $P_2$ .
2.  $P_1$  dan  $P_2$  secara simultan mengkomunikasikan  $x$  dengan  $P_3$  dan  $P_4$
3.  $P_1, P_2, P_3$  dan  $P_4$  secara simultan meng-komunikasikan  $x$  dengan  $P_5, P_6, P_7$  dan  $P_8$  .

Dan seterusnya hingga semua prosesor mengenal  $x$ . Proses ini dilakukan dalam  $\log N$  langkah.

# Analisa Algoritma Paralel

➤ Searching, File dimana  $x$  akan dicari dibagi ke dalam sub file dan secara simultan dilakukan pencarian oleh prosesor

prosesor :

- $P_1$  mencari pada  $n/N$  entri pertama,
- $P_2$  mencari pada  $n/N$  entri kedua,
- $P_3$  mencari pada  $n/N$  entri ketiga,
- $P_N$  mencari pada  $n/N$  entri ke-N.

Proses ini membutuhkan  $n/N$  langkah.

Jadi total langkah yang dibutuhkan oleh algoritma tersebut

adalah :  $\log N + n/N$  langkah.

# Analisa Algoritma Paralel

## Speedup

Pengukuran speedup sebuah algoritma paralel adalah salah satu cara untuk mengevaluasi kinerja algoritma tersebut.

Speedup adalah perbandingan antara waktu yang diperlukan algoritma sekuensial yang paling efisien untuk melakukan komputasi dengan waktu yang dibutuhkan untuk melakukan komputasi yang sama pada sebuah mesin pipeline atau paralel.

**Worst case running time dari algoritma sekuensial terefisien**

$$\text{Speedup} = \frac{\text{Worst case running time dari algoritma sekuensial terefisien}}{\text{Worst case running time dari algoritma paralel}}$$

# Analisa Algoritma Paralel

## ➤ Banyaknya Prosesor

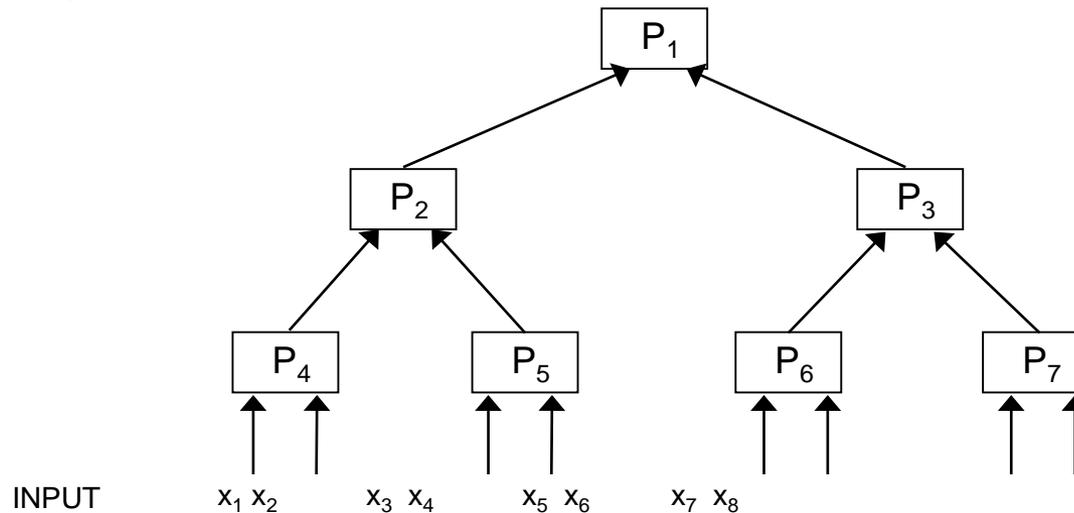
Semakin banyak prosesor yang digunakan semakin tinggi biaya untuk memperoleh solusi sebuah problem. Hal ini terjadi karena perlu dipertimbangkan biaya pengadaan prosesor dan perawatannya. Jumlah prosesor yang tergantung dari  $n$ ,  $n$ =ukuran problem, dinyatakan sebagai  $p(n)$ . Kadang-kadang jumlah prosesor tidak tergantung pada ukuran problem.

# Analisa Algoritma Paralel

Contoh 3 :

Perhatikan  $n$  bilangan  $x_1, x_2, \dots, x_n$  yang akan dijumlahkan. Dengan menggunakan komputer tree-connected SIMD dengan  $\log n$  level dan  $n/2$  daun, dibutuhkan pohon dengan ukuran  $(n-1)$  atau  $p(n) = n - 1$ .

Ilustrasi untuk  $n = 8$ . Sedangkan pada contoh 1., banyaknya prosesor,  $N$ , tidak tergantung pada Ukuran problem,  $n$ .



TERIMAKASIH

# Kuis Komputer Paralel – Pertemuan ke-15

1. Jelaskan fungsi Cluster dalam interkoneksi sistem komputer dan apa manfaatnya?
2. Bagaimana performa kinerja dari komputer paralel yang disusun dari spesifikasi yang berbeda ?

*Note: Kerjakan di selembar kertas/buku catatan mata kuliah Arsikom kalian!*

