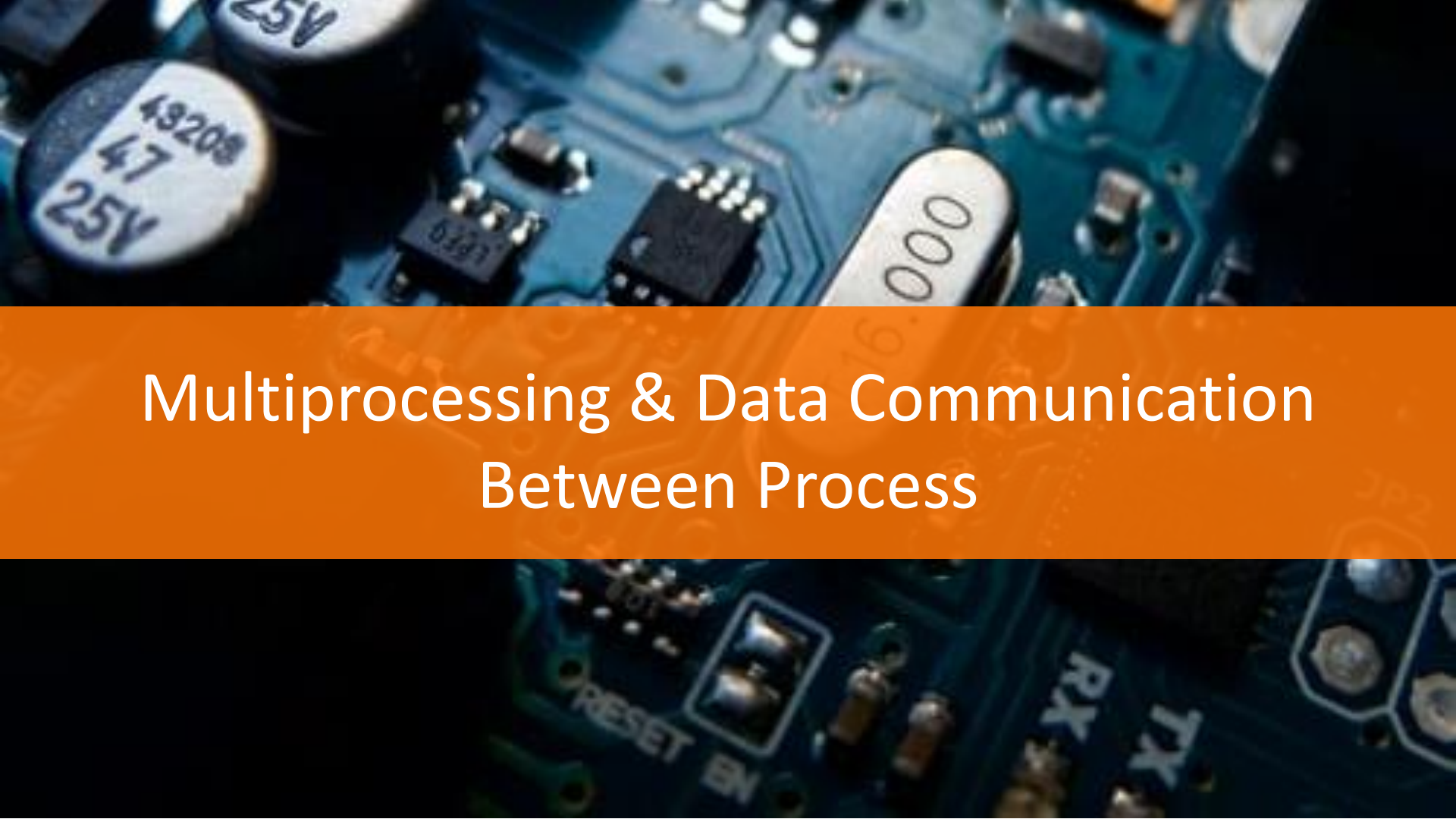
PEMROSESAN PARALEL

CCE60218

Teknik Komputer

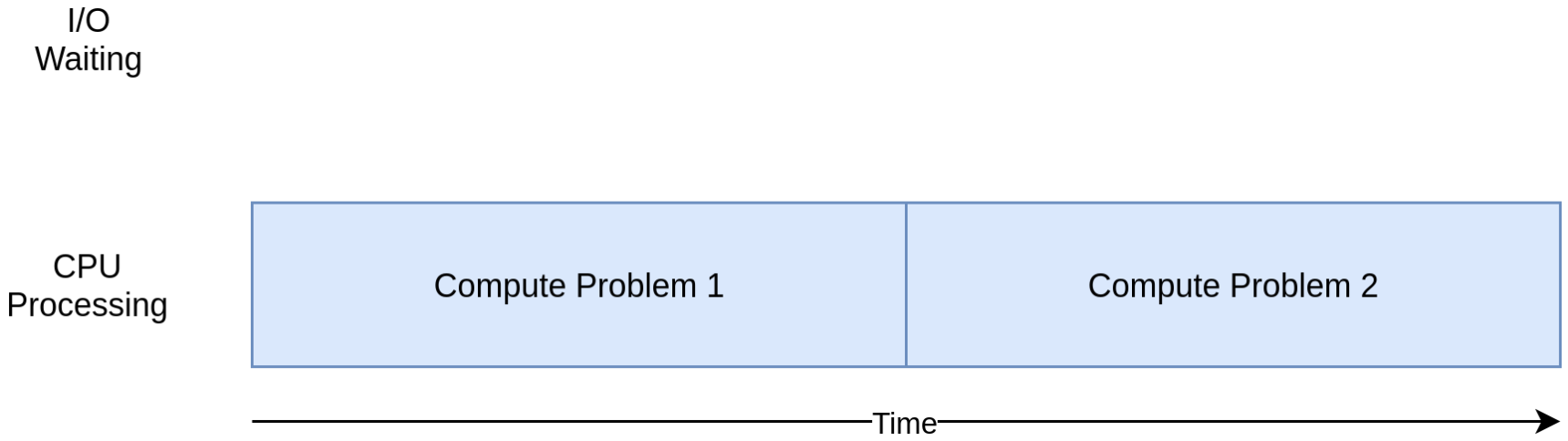


Multiprocessing & Data Communication Between Process

Multiprocessing

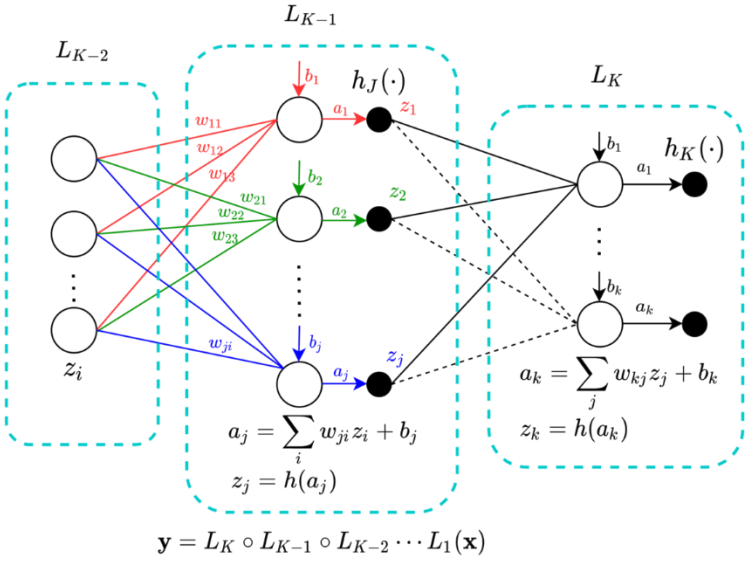
- Multiprocessing refers to the ability of a system to support more than one processor at the same time. Applications in a multiprocessing system are broken to smaller routines that run independently.

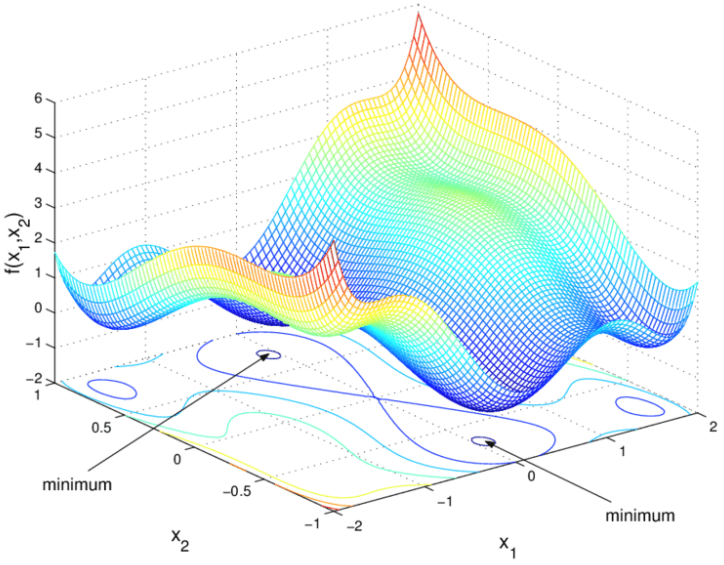
CPU-Bound Process



Example

- ANN Training (Back Propagation)
- Numerical Optimization
- Matrices Multiplication





$$\begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix} \cdot \begin{pmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{pmatrix} \rightarrow \begin{pmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{pmatrix}$$

Task 1: $C_{1,1} = A_{1,1}B_{1,1} + A_{1,2}B_{2,1}$

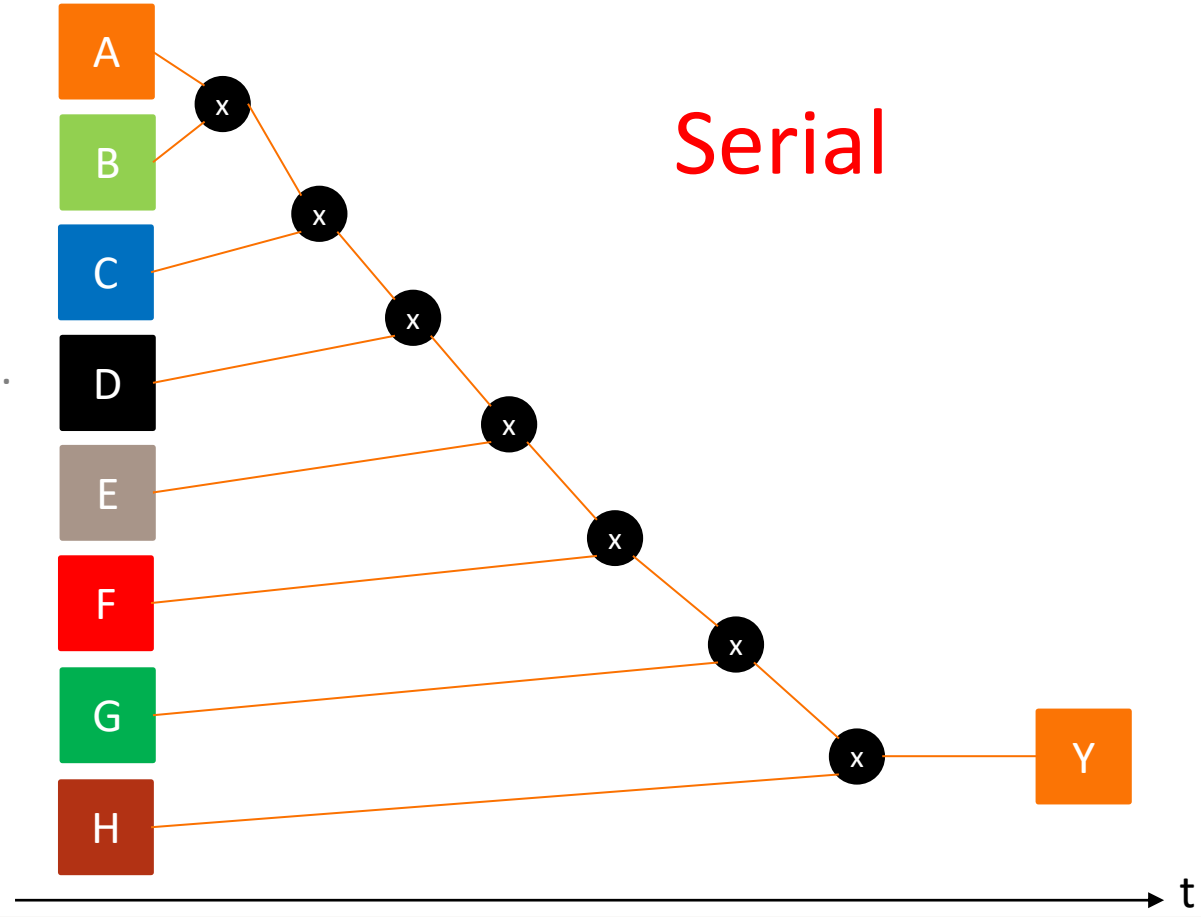
Task 2: $C_{1,2} = A_{1,1}B_{1,2} + A_{1,2}B_{2,2}$

Task 3: $C_{2,1} = A_{2,1}B_{1,1} + A_{2,2}B_{2,1}$

Task 4: $C_{2,2} = A_{2,1}B_{1,2} + A_{2,2}B_{2,2}$

Example

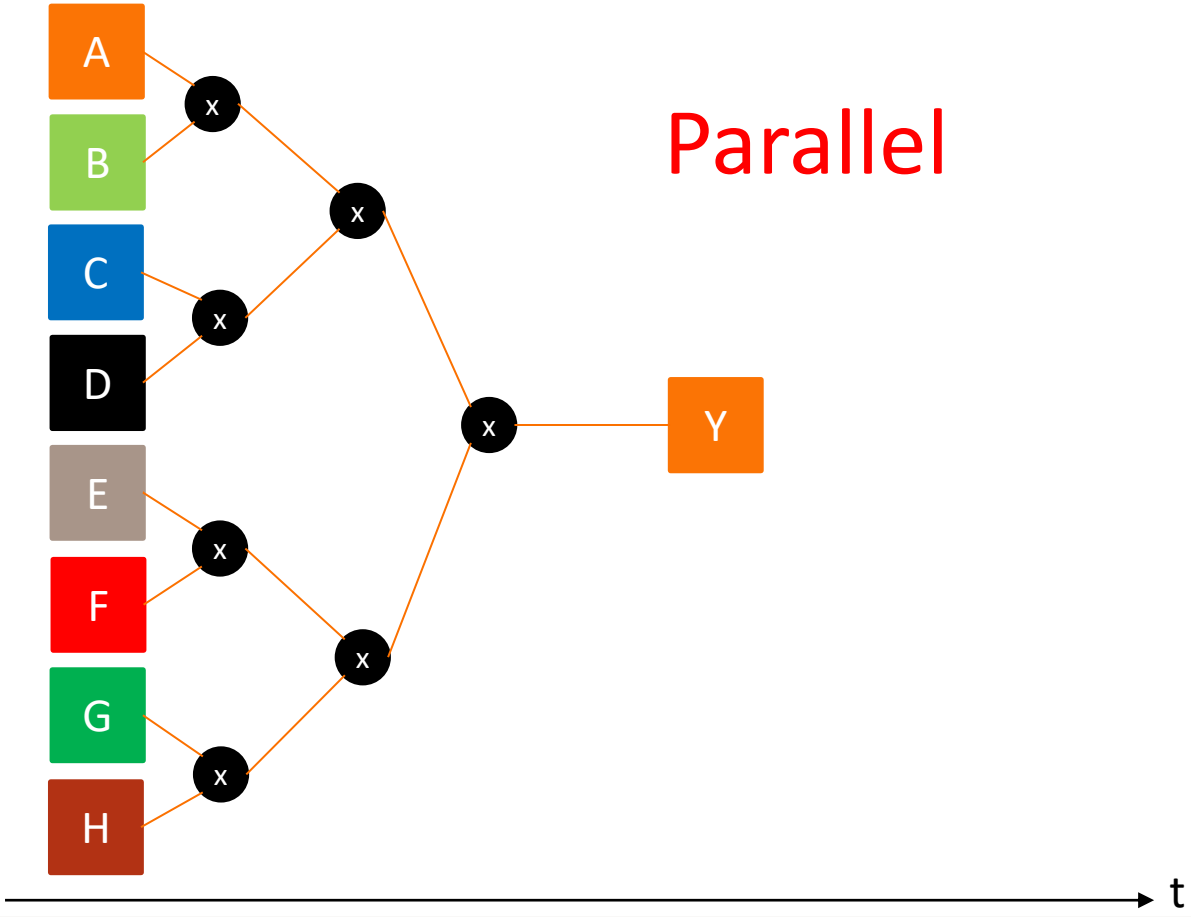
Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:
Y = A x B x C x D x E x F x G x H.
dimana
Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5



Serial

Example

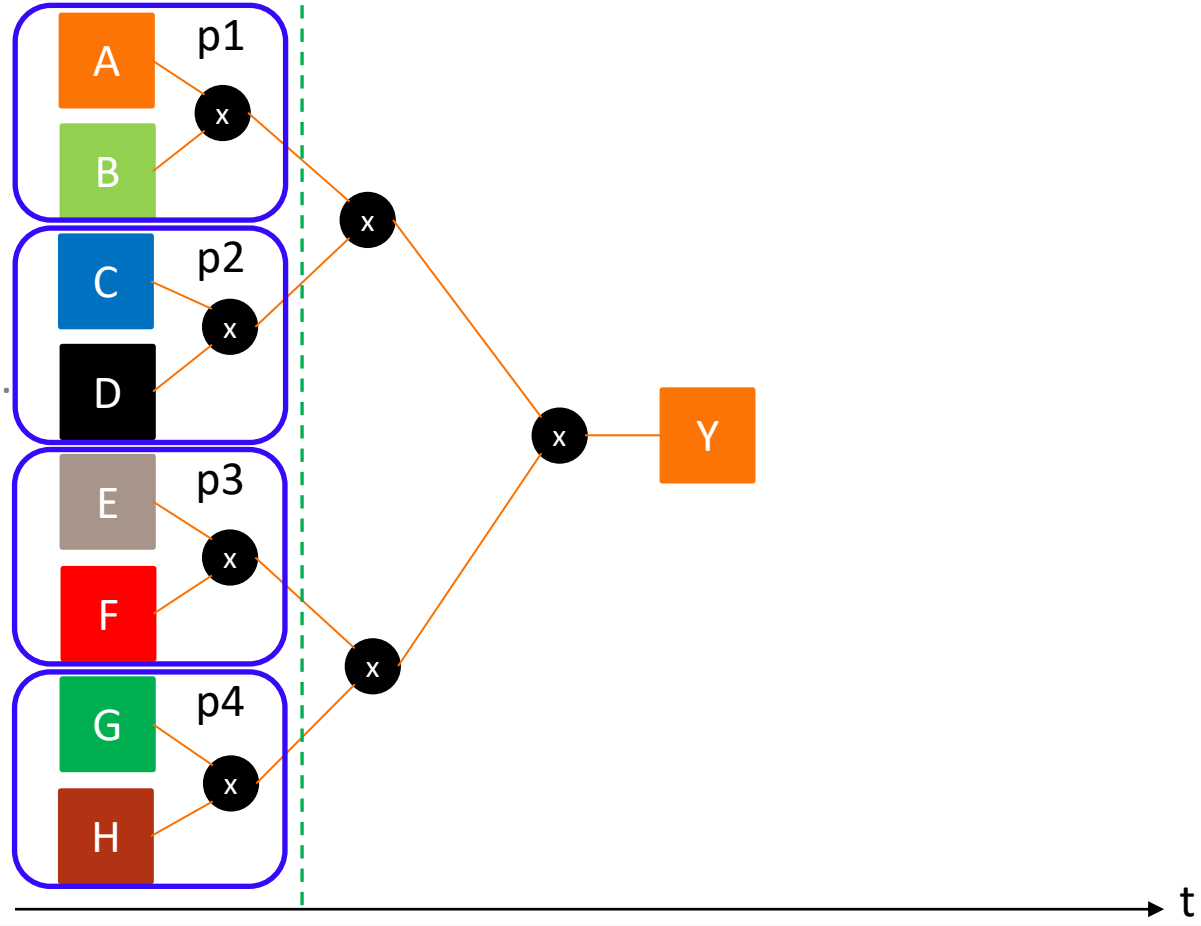
Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:
Y = A x B x C x D x E x F x G x H.
dimana
Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5



Parallel

Example

Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:
Y = A x B x C x D x E x F x G x H.
dimana
Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5



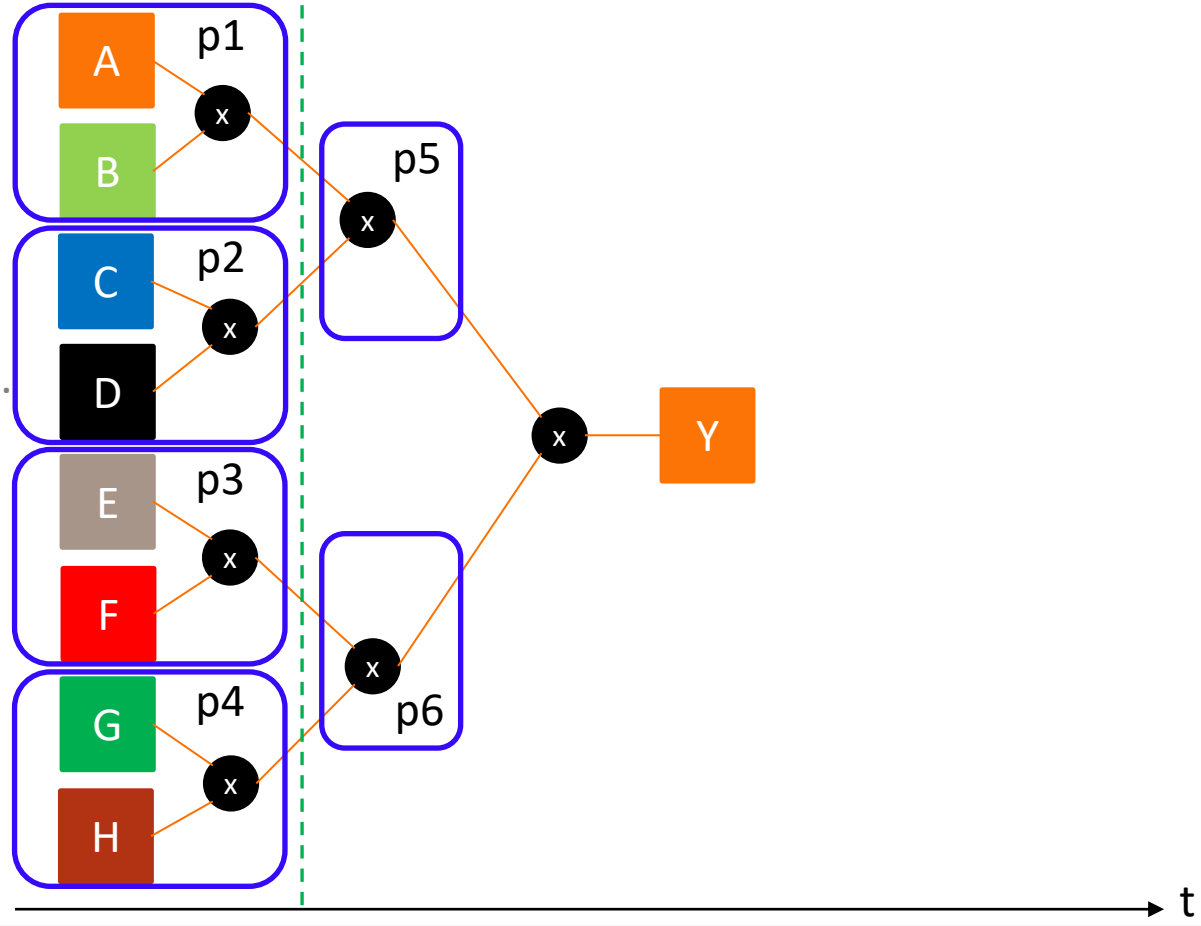
Example

Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:

Y = A x B x C x D x E x F x G x H.

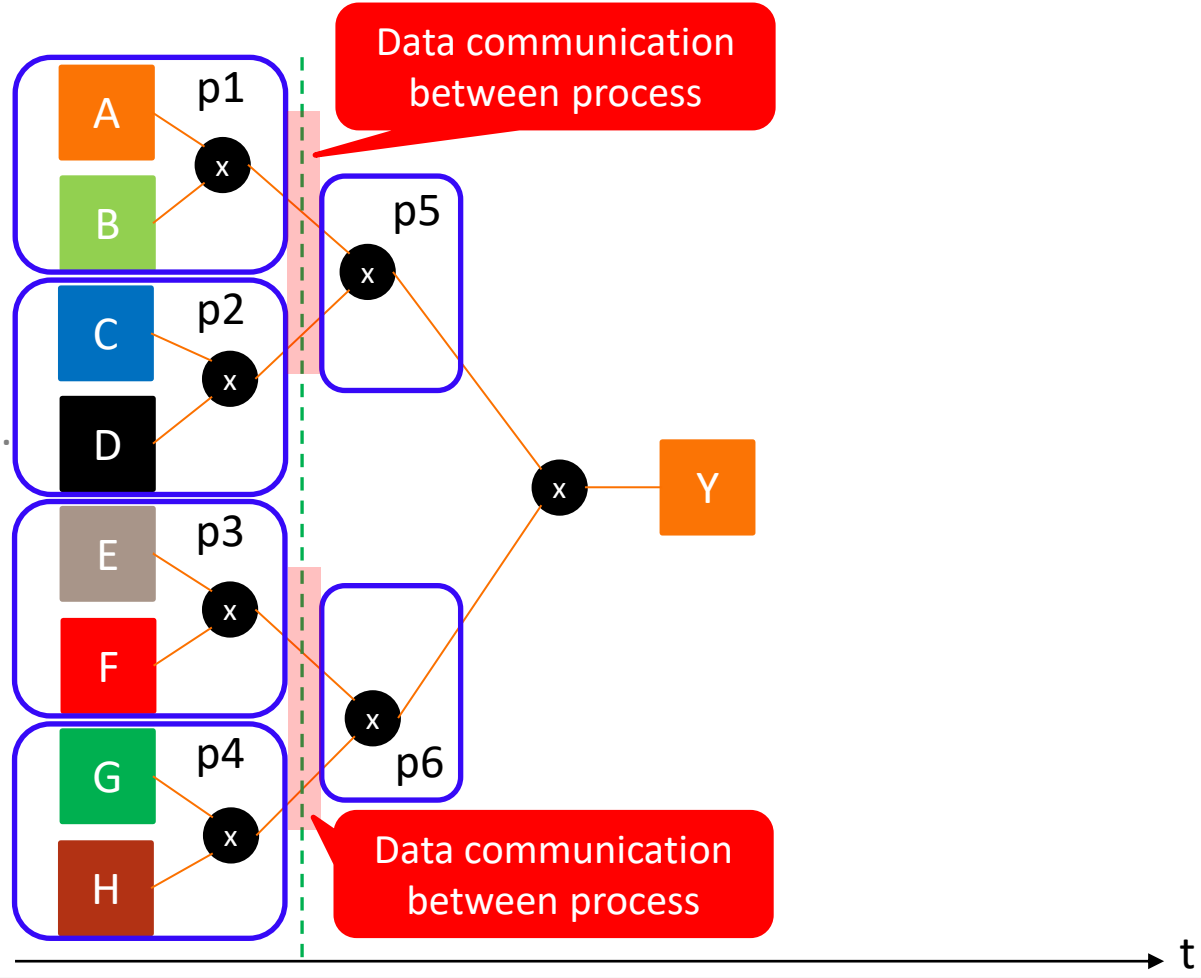
dimana

Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5



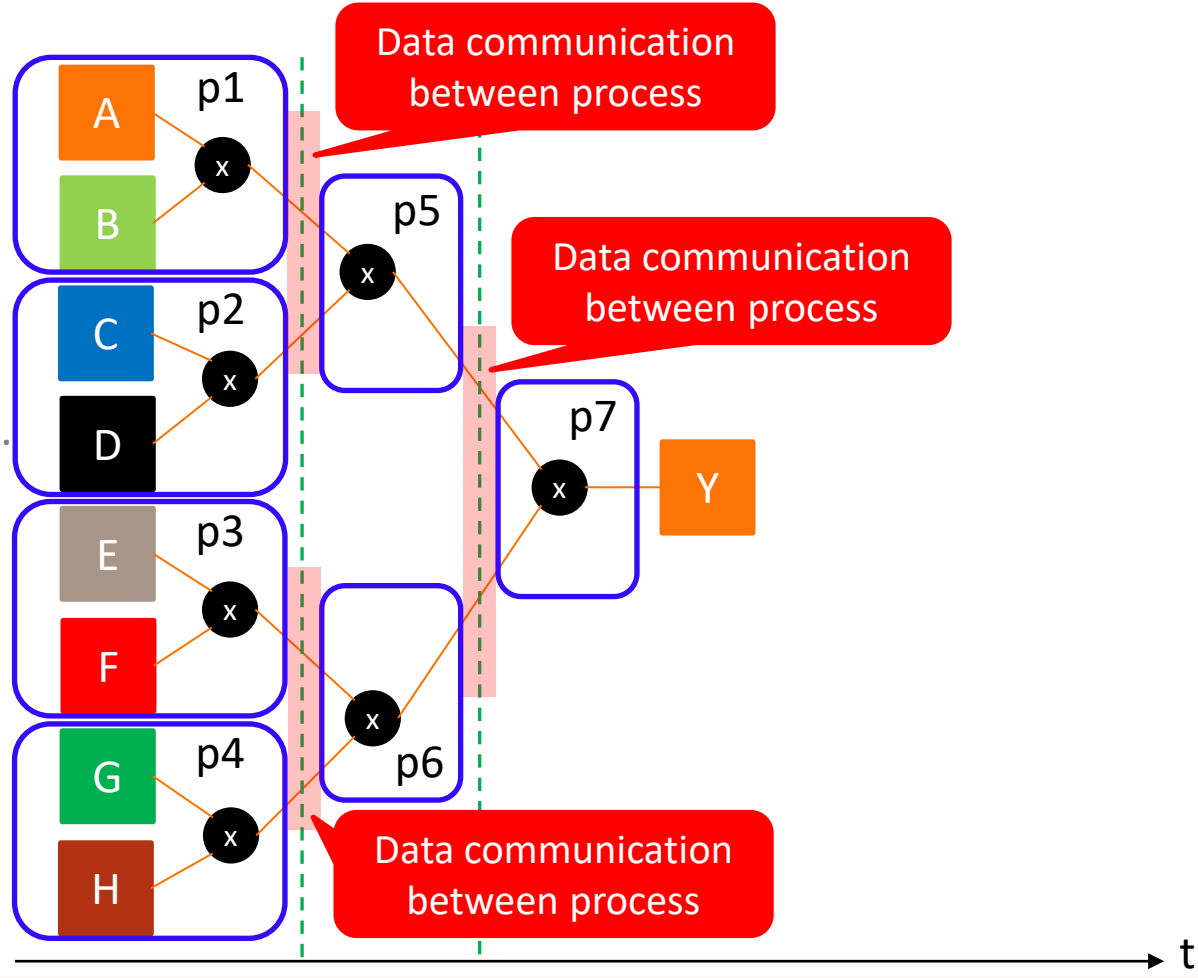
Example

Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:
Y = A x B x C x D x E x F x G x H.
dimana
Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5



Example

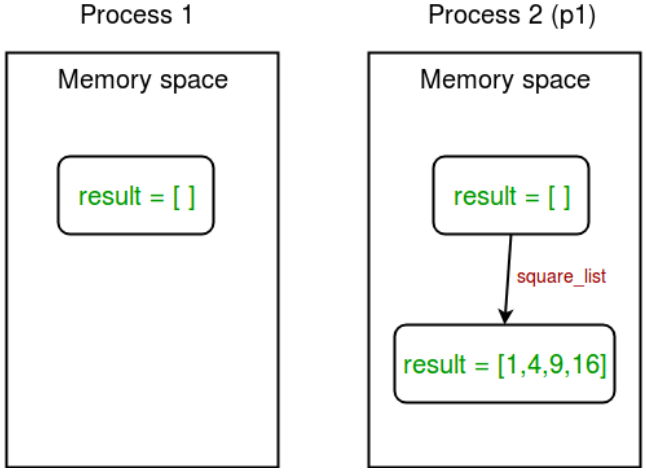
Buatlah algoritma dalam bentuk pseudococe untuk chained matrices multiplication:
Y = A x B x C x D x E x F x G x H.
dimana
Y, A, B, C, D, E, F, G, dan H adalah matriks 5 x 5

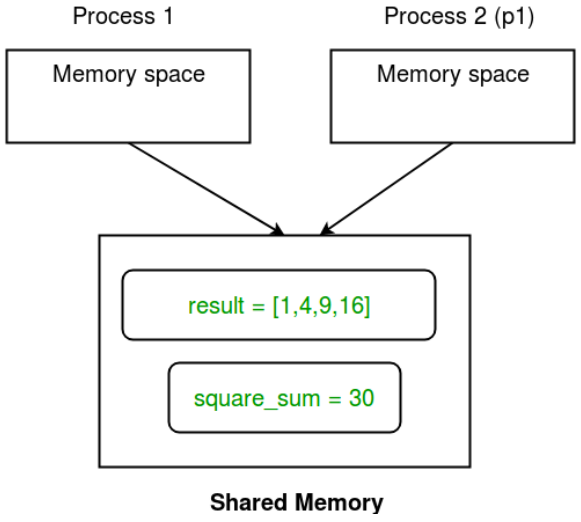


Data Communication Between Process

Shared Memory: multiprocessing module provides **Array** and **Value** objects to share data between processes.

- **Array:** a ctypes array allocated from **shared memory**: `multiprocessing.Array('i', 4)`
- **Value:** a ctypes object allocated from **shared memory**: `multiprocessing.Value('i')`

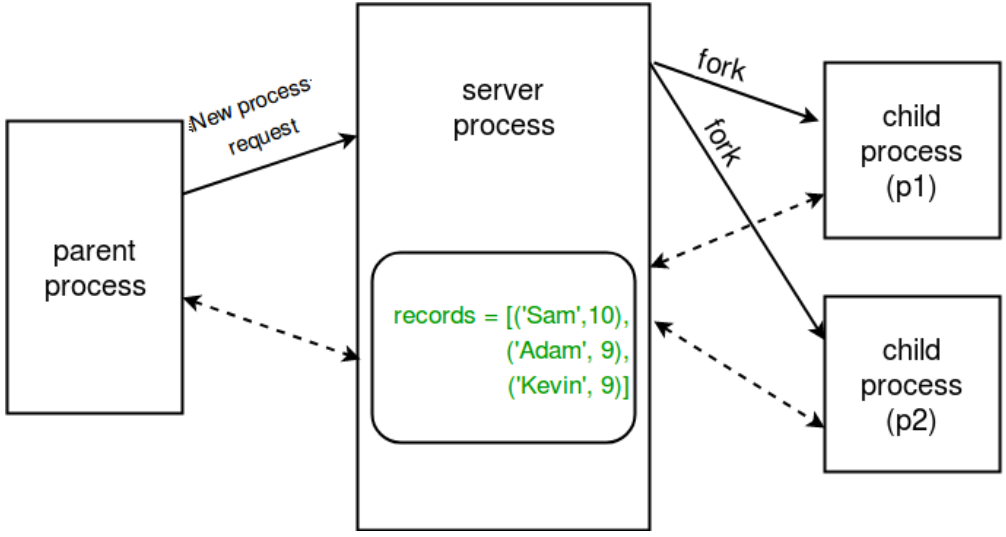




Data Communication Between Process

Server process : Whenever a python program starts, a **server process** is also started. From there on, whenever a new process is needed, the parent process connects to the server and requests it to fork a new process.

A **server process** can hold Python objects and allows other processes to manipulate them using proxies.



Terima Kasih

Teknik Komputer