

ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

---

## THIẾT KẾ VÀ ĐÁNH GIÁ THUẬT TOÁN

### Bài 9

# Giới thiệu tính toán song song (*Introduction to Parallel Computing*)

*Nguyễn Thị Hồng Minh*

*minhnth@gmail.com*

# Nội dung

- 1. Giới thiệu về xử lý song song**
- 2. Hệ thống máy tính song song**
- 3. Thuật toán song song**

# Giới thiệu về xử lý song song

- Tại sao phải xử lý song song (XLSS)?
- XLSS là gì ?
- Những ứng dụng chính.
- Các vấn đề liên quan trong nghiên cứu XLSS.

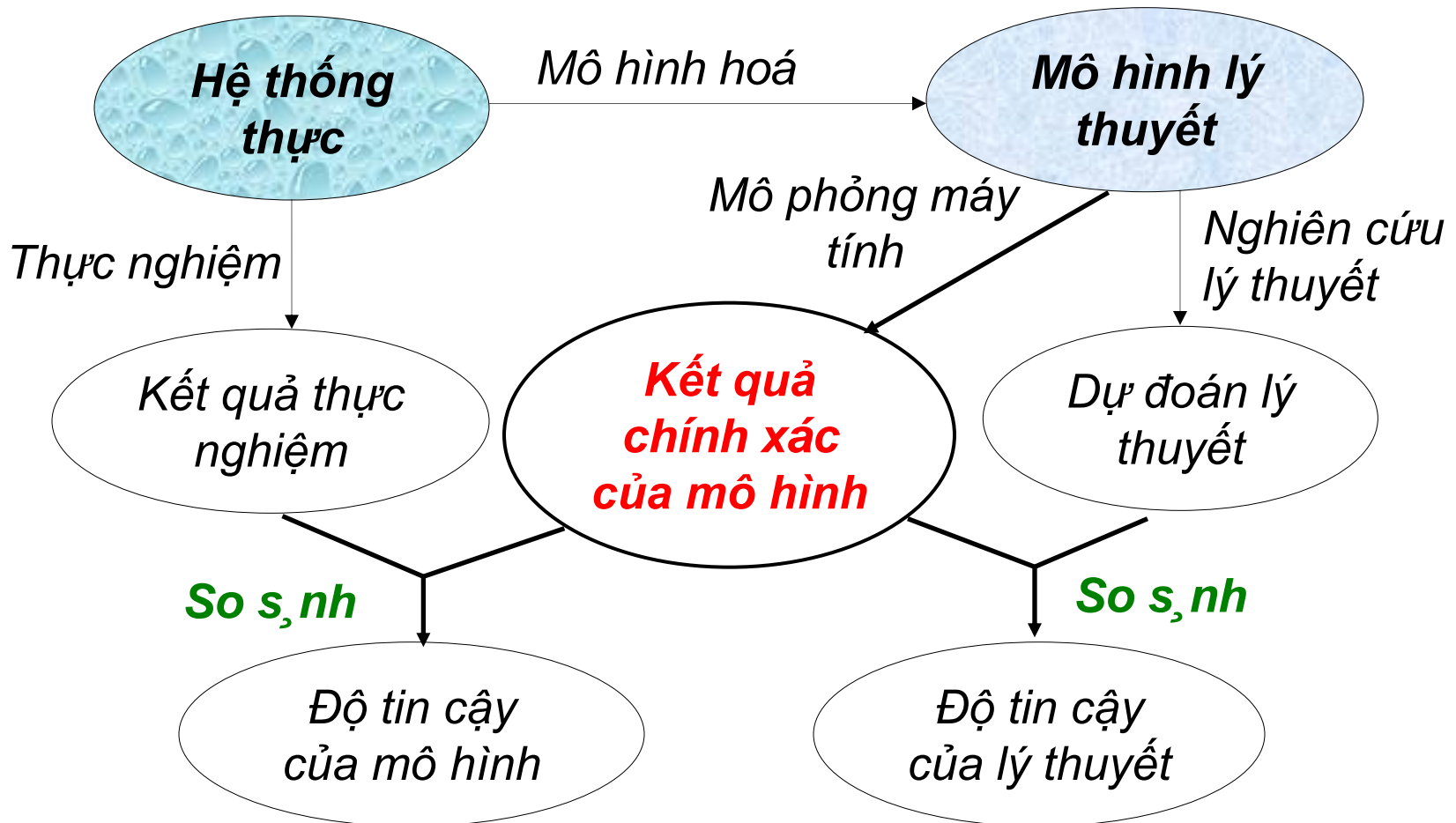
# Giới thiệu về xử lí song song

- **Tại sao phải xử lí song song**

- Yêu cầu tính toán ngày càng lớn: thực tại ảo, cơ sở dữ liệu phân tán, trí tuệ nhân tạo, mô hình mô phỏng, khai phá dữ liệu...
- Nhiều các ứng dụng cần tính toán với tốc độ peta-flop (chúng ta mới đang đạt tốc độ tera-flop)
- Mô phỏng (*simulation*), một hướng nghiên cứu mới mẻ của khoa học. Những bài toán xử lí hàng triệu các đối tượng giả định.
- Những kiến trúc tuần tự khó phát triển mạnh hơn nữa. Giá thành của các hệ thống song song đang ngày càng giảm mà sức mạnh đang ngày càng tăng

# Giới thiệu về xử lý song song

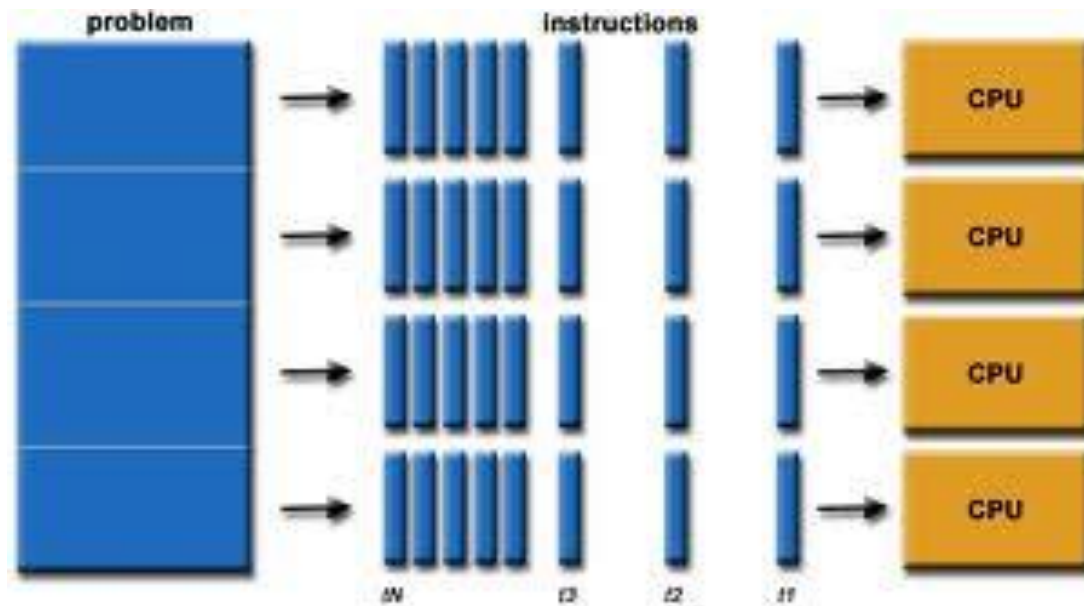
- Mô phỏng bằng máy tính



# Giới thiệu về xử lý song song

- **Xử lý song song là gì?**

- Nhiều “bộ xử lý” đồng thời phối hợp thực hiện các tiến trình để giải quyết cùng một bài toán.



- Sự liên kết và đồng bộ giữa các tiến trình là những vấn đề cơ bản của lập trình song song.

# Giới thiệu về xử lí song song

- **Ứng dụng của xử lí song song**

- **Tính toán khoa học và kĩ thuật**

- Mô hình toán học: khí hậu, dòng chảy, sơ đồ gen...
    - Thiết bị không gian
    - Siêu vật liệu
    - ...

- **Thương mại**

- Máy chủ web và máy chủ dữ liệu lớn:
      - ✱ [eBay](#) có data warehouse cỡ 6.5 petabyte
      - ✱ [Facebook](#) có data warehouse cỡ 2.5 petabyte  
<http://sceweb.uhcl.edu/boetticher/CSCI5931%20Big%20Data.html>

- **Hệ thống máy tính và mô phỏng**

- Trí tuệ nhân tạo, người máy
    - Hệ không gian, vũ trụ...

# Giới thiệu về xử lý song song

- Theo Top 500 Supercomputer site: Application Areas

Chess	Heavy Industry	Services
Aerospace	Image Processing/Rendering	Software
AFIS	Information Service	Telecommunication
Automotive	Information Processing Service	Transportation
Benchmarking	Life Science	Utility
Biology	Manufacturing	Weather and Climate Research
Biochemistry	Mechanics	Web Services
CFD	Medicine	Weather Forecasting
Chemistry	Media	Digital Content Creation
Construction	Pharmaceutics	Semiconductor
Consulting	Research	Digital Media
Database	Services	Entertainment
Defense	Software	Not Specified
Economics	Telecommunication	Gaming
Electronics	Transportation	Internet Provider
Energy	Utility	Retail
Environment	Weather and Climate Research	Logistic Services
Finance	Web Services	Cloud Services
Geophysics	Weather Forecasting	IT Services
Hardware	Digital Content Creation	Insurance Services



# Hệ thống xử lí song song

- **Phần cứng**

- Các thành phần của máy tính song song
- Phân loại máy tính song song

- **Phần mềm**

- Hệ điều hành song song
- Ngôn ngữ lập trình song song

# Phần cứng

- **Các thành phần của máy tính song song**

- Kiến trúc logic:

- Cơ cấu điều khiển: Cơ chế điều khiển hoạt động của bộ nhớ, bộ xử lý.
    - Mô hình liên kết: Phương thức tạo liên kết giữa các bộ xử lý với nhau và bộ xử lý với bộ nhớ
      - ✳ Mô hình không gian địa chỉ chia sẻ
      - ✳ Mô hình truyền thông điệp

- Kiến trúc vật lý: Mô hình máy tính thực tế

- Bộ xử lý
    - Bộ nhớ: bộ nhớ truy cập ngẫu nhiên song song (PRAM), phân tán
    - Mạng liên kết trong: Tạo các liên kết giữa các bộ xử lý với nhau và các bộ xử lý với bộ nhớ (mạng tuyến tính và vòng; mạng liên kết lưới, sao; phi tuyến...)

# Phần cứng

- **Phân loại các hệ thống máy tính song song**
  - Máy tính song song chuyên dụng (Parallel Special Purpose Computer)
    - Máy tính được thiết kế với các chip hỗ trợ cùng bộ xử lý chính thực hiện một mục đích tính toán cụ thể.
    - Ví dụ: MDGRAPE tính toán mô phỏng động lực phân tử (Molecular Dynamic Simulation – MDS)

System	Peak speed (Gflops)	Cost/speed (\$/Gflops)	Power/speed (Watt/Gflops)	Size/speed (liter/Gflops)
<b>MDGRAPE-3</b>	1,000,000	<b><u>9</u></b>	<b><u>0.2</u></b>	<b><u>0.03</u></b>
Blue Gene/L	360,000	280	4	0.3
PC (Pentium D 3.0GHz)	12	80	25	3

(Nguồn: Tetsu Narumi – NIC – 2006)

# Phần cứng

- **Phân loại các hệ thống máy tính song song**
  - Siêu máy tính (supercomputer)
    - Hệ thống với nhiều bộ xử lý, nhiều bộ nhớ được tổ chức trong cùng một kiến trúc vật lý.
    - Ưu điểm: Tốc độ tính toán lớn
    - Hạn chế: Giá thành cao
    - Một số thông số đánh giá siêu máy tính:
      - ✦ Số bộ xử lý: (cores)
      - ✦ Tốc độ (Tflop/s): thiết kế (Rpeak), tối đa đạt được (Rmax)
      - ✦ Năng lượng điện tiêu thụ (kW)
      - ✦ Lĩnh vực ứng dụng,...

# Phần cứng

## Top 500 supercomputers – 11/2012

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	<b>Titan</b> - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
2	DOE/NNSA/LLNL United States	<b>Sequoia</b> - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	16,324.8	20,132.7	7,890
3	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
4	DOE/SC/Argonne National Laboratory United States	<b>Mira</b> - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,162.4	10,066.3	3,945
5	Forschungszentrum Juelich (FZJ) Germany	<b>JUQUEEN</b> - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM	393,216	4,141.2	5,033.2	1,970

*Nguồn [www.top500.org](http://www.top500.org)*

# Phần cứng

## Top 500 supercomputers – 11/2013

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Super Computer Center in Guangzhou China	<b>Tianhe-2 (MilkyWay-2)</b> - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
2	DOE/SC/Oak Ridge National Laboratory United States	<b>Titan</b> - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
3	DOE/NNSA/LLNL United States	<b>Sequoia</b> - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
4	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
5	DOE/SC/Argonne National Laboratory United States	<b>Mira</b> - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,586.6	10,066.3	3,945

*Nguồn [www.top500.org](http://www.top500.org)*

# Phần cứng

- **Phân loại các hệ thống máy tính song song**

- **Cụm máy tính (cluster)**

- Hệ thống song song dựa trên kết nối mạng các trạm (node) làm việc
    - Ưu điểm: Phù hợp với nhiều ứng dụng; Dễ mở rộng; Độ tin cậy cao; Giá thành hạ; có HW/SW chuẩn.
    - Nhược điểm: Tốc độ không đạt chuẩn thiết kế; Độ trễ cao do sử dụng đường truyền vật lý
    - Ví dụ: Clusters tại HPC-HUS
      - ✳ IBM 1600 p-series: 5 node tính toán, năng lực tính toán: 240 Gflops (mở rộng tối đa 768 Gflops/16 node).
      - ✳ IBM 1350 x-series: 8 node tính toán, năng lực tính toán: 51,2 Gflops, cho phép thực hiện tính toán từ xa (qua mạng).

# Phần mềm

- **Hệ điều hành**
- **Ngôn ngữ lập trình**
- **Phần mềm ứng dụng**



# Phần mềm

- **Hệ điều hành**

- Chức năng:

- Khởi động hệ thống
    - Phân đoạn chương trình và lập lịch cho các tiến trình
    - Trao đổi và đồng bộ hóa các tiến trình
    - Quản lý và điều hành hệ thống

- Ba tổ chức cơ bản của HĐH đa xử lý:

- Tổ chức khách-chủ
    - Tổ chức thực thi độc lập
    - Tổ chức đối xứng

# Phần mềm

- **Ngôn ngữ lập trình**

- **Một số mô hình lập trình song song**

- **Task/Channel** : Các tác nhiệm (*task*) thực hiện trên mỗi BXL; Các tín hiệu trao đổi giữa các PE thông qua các kênh truyền (*channel*)
    - **Message passing**: Các tác nhiệm được đặt tên; Trao đổi giữa các tác nhiệm thông qua truyền thông điệp.
    - **Share memory**: Các tác nhiệm chia sẻ không gian địa chỉ chung, đọc/ghi không đồng bộ..
    - **Data Parallelism**: Khai thác sự đồng thời của các tác nhiệm khác nhau trên phân hoạch của các dữ liệu lên các BXL

# Phần mềm

- **Ngôn ngữ lập trình**
  - **Một số ngôn ngữ/thư viện lập trình song song**
    - **Fortran 90**: Mô hình Data Parallelism
    - **OCCAM**: Mô hình Task and Channel
    - **nCUBE C**: Mô hình Share memory
    - **MPI (Message Passing Interface)**: Mô hình truyền thông điệp

# Phần mềm

- **Ngôn ngữ lập trình**

- **MPI:** thư viện hỗ trợ lập trình song song dựa trên cơ chế truyền thông điệp.
  - Được thiết kế để chạy cùng các phần mềm trung gian (C hoặc Fortran).
  - Các tiến trình được liên kết thông qua các thao tác truyền thông điệp hiệu quả: tránh sao chép bộ nhớ và cho phép tính toán và truyền thông đồng thời.
  - Cơ chế tự động đồng bộ các giao tác truyền thông.
  - Hỗ trợ môi trường không thuần nhất.

# Thuật toán song song

- **Vấn đề cơ bản trong thiết kế thuật toán song song**
  - Phân rã được các tác nhiệm có thể thực hiện song song.
  - Tổ chức truyền thông giữa các tác nhiệm.
  - Đồng bộ hóa dữ liệu, thời gian thực hiện giữa các tác nhiệm.
- Ví dụ: Thuật toán tính tổng các phần tử dãy số  $(a,n)$ :  
    for i = 1 to n do  
        sum = sum + a[i]  
    End;

Có thể song song hóa được?

# Thuật toán song song

- **Vấn đề cơ bản trong thiết kế thuật toán song song.**

- Ví dụ: Thuật toán tính tổng các phần tử dãy số  $(a, n)$ :

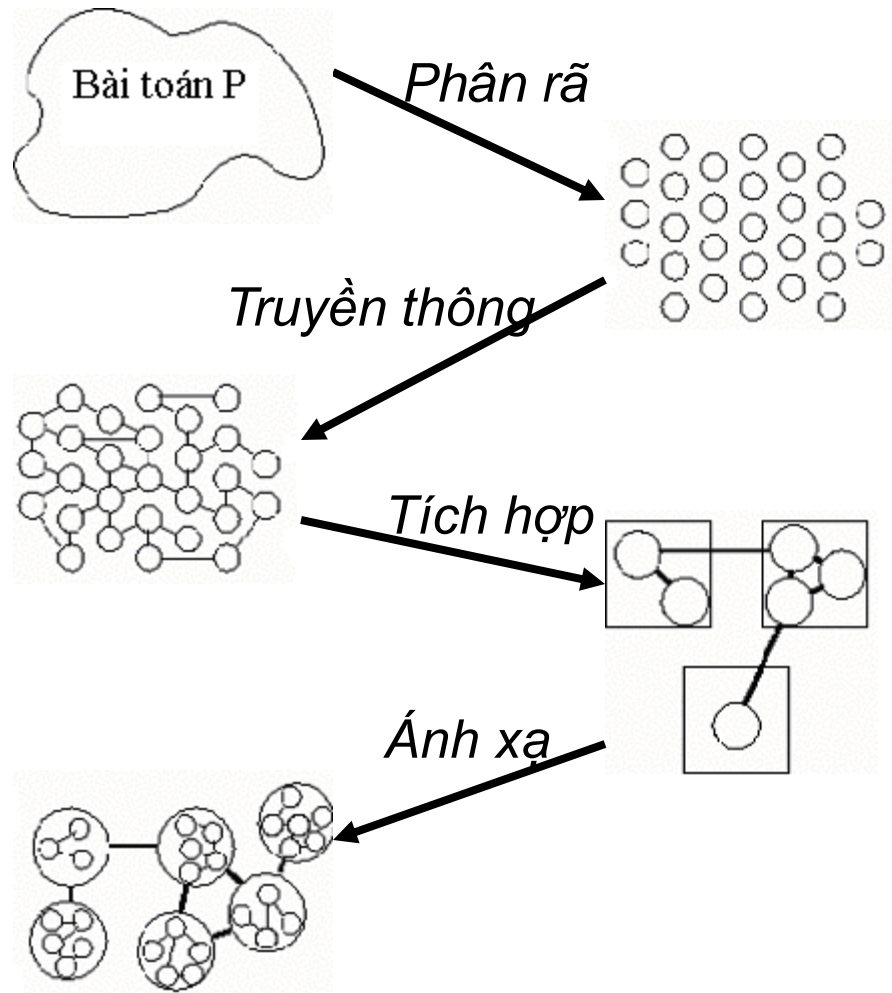
```
for i = 1 to n do  
    sum = sum + a[i]  
end;
```

- Thay đổi/phân rã lại tính toán:

```
for i = 0 to n/4 do  
    s1 = s1 + a[i*4+1]; s2 = s2 + a[i*4+2];  
    s3 = s3 + a[i*4+3]; s4 = s4 + a[i*4+4];  
end;  
sum = s1 + s2 + s3 + s4;
```

# Thuật toán song song

- Các giai đoạn của quá trình thiết kế TTSS



Phân tích các tính toán thành các tác vụ nhỏ



Thiết lập cấu trúc truyền thông phối hợp các tác vụ



Tích hợp một số tác vụ thích hợp cho mỗi BXL



Đặt các tác vụ có khả năng chạy song song lên các bộ xử lý khác nhau

# Thuật toán song song

- **Đánh giá độ phức tạp của thuật toán song song**
  - **Độ phức tạp thời gian**
    - Đánh giá độ phức tạp thời gian theo quan hệ tiệm cận  $O(f(x))$ ,  $\Theta(f(x))$ ,  $\Omega(f(x))$
  - **Độ phức tạp không gian:**
    - Bộ nhớ
    - Bộ xử lý
  - **Giá (chi phí) của thuật toán**
    - Tổng các phép tính toán được thực hiện trên tất cả các bộ xử lý
    - Ví dụ: Một thuật toán chạy trên mô hình PRAM có độ phức tạp thời gian  $\Theta(\log n)$  sử dụng  $p$  bộ xử lý sẽ có giá là  $\Theta(p \log n)$ .



# Thuật toán song song

- **Đánh giá thuật toán song song**
  - **Hệ số tăng tốc (speed up):** Tỷ lệ giữa thời gian của thuật toán tuần tự tốt nhất và thời gian thực hiện song song trên hệ thống song song với  $p$  bộ xử lý

$$S_p = t_s / t_p \quad (1 \leq S_p \leq p)$$

- **Hiệu suất (efficiency):** Hệ số tăng tốc trên mỗi bộ xử lý

$$E_p = S_p / p = t_s / p t_p \quad (E_p \leq 1)$$

- **Giá (cost):**  $C_p = p t_p$

$t_s$  thời gian thực hiện thuật toán tuần tự

$t_p$  thời gian thực hiện thuật toán song song trên  $p$  bộ xử lý

# Thuật toán song song

- **Ví dụ**

- **Bài toán tìm kiếm trên mảng được sắp:**

Cho một dãy được sắp  $X = (x_1, x_2, x_3, \dots, x_n)$ ,  $x_i < x_{i+1} \quad \forall x_i \in S$

Với phần tử  $y$  cho trước thuộc  $S$ , tìm chỉ số  $i$  sao cho

$$x_i \leq y < x_{i+1} \text{ với } x_0 = -\infty \text{ và } x_{n+1} = +\infty$$

- **Tìm kiếm nhị phân:** Độ phức tạp  $O(\log n)$  trên 1 bộ xử lý
  - Nếu có  $p$  bộ xử lý?

# Thuật toán song song

- Thuật toán tìm kiếm song song trên  $p$  bộ xử lý
- Ý tưởng: Áp dụng tư tưởng của thuật toán tìm kiếm nhị phân.
  - Chia  $X$  thành  $p+1$  đoạn bằng nhau sau đó
  - Mỗi bộ xử lý sẽ thực hiện nhận dạng  $x_i$  nằm trên đoạn mình phụ trách
  - Xác định  $X$  mới theo chỉ số của đoạn tìm được, tiếp tục chia để trị



- Thuật toán thực hiện trên bộ xử lý thứ  $j$  ( $j=1..p$ )

**Begin**

1. if ( $j = 1$ ) then

1.1  $\{ l := 0; \quad r := n + 1;$   
 $x_0 := -\infty; \quad x_{n+1} := +\infty;$   
 $c_0 := 0; \quad c_{p+1} := 1; \}$

# Thuật toán song song

```
2.  While  $(r - l > p)$  do {  
2.1      if  $(j = 1)$  then  $\{ q_0 := l, q_{p+1} := r \}$   
2.2       $q_j := l + j * \lceil r - l / p + 1 \rceil$       // chỉ số điểm đầu của mỗi đoạn  
2.3      if  $(y = x_{q_j})$  then  $\{ \text{return } (q_j); \text{exit} \}$   
          else  $\{ \text{if } y > x_{q_j} \text{ then } c_j := 0 \text{ else } c_j := 1 \}$   
2.4      if  $(c_j < c_{j+1})$  then  $\{ l := q_j; r := q_{j+1} \}$   
2.5      if  $(j = 1 \text{ and } c_0 < c_1)$  then  $\{ l := q_0; r := q_1 \}$  //đoạn đầu tiên  
          }      //End While
```

```
3.  if  $(j \leq r - l)$  then
```

```
3.1      Case {  
           $y = x_{l+j} : \{ \text{return } (l + j); \text{exit} \}$   
           $y > x_{l+j} : c_j := 0$   
           $y < x_{l+j} : c_j := 1 \}$ 
```

```
3.2      if  $(c_{j-1} < c_j)$  then return  $(l + j - 1)$ 
```

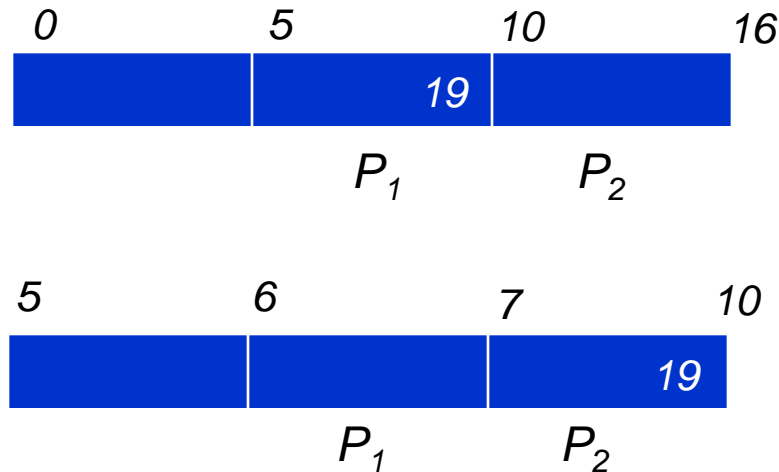
**End.**

# Thuật toán song song

Minh họa: Xét dãy  $X = (2, 4, 6, \dots, 30)$ ;  $y = 19$ ;  $p = 2$ ;

Bước 1 (trên  $P_1$ ):  $l = 0$ ,  $r = 16$ ,  $c_0 = 0$ ,  $c_3 = 1$ ,  $x_0 = -\infty$ ,  $x_{16} = +\infty$ .

Vòng lặp while (2) thực hiện 3 lần. Bước 2.1 chỉ thực hiện trên bộ xử lý  $P_1$ , thu được  $q_0 = 0$ ,  $q_3 = 16$  các bước còn lại thực hiện trên cả hai bộ xử lý. Sau 3 lần lặp giá trị của các biến như sau:



Lần lặp	1	2	3
$q_0$	0	5	7
$q_1$	5	6	8
$q_2$	10	7	9
$q_3$	16	10	10
$c_1$	0	0	0
$c_2$	1	0	0
$c_3$	1	1	1
$l$	5	7	9
$r$	10	10	10

Bước 3.1 và 3.2 thực hiện trên bộ xử lý  $p_1$  và  $c_1 = 1$  như vậy  $c_0 < c_1$ .

Kết quả trả về là 9 (giá trị  $x_9 = 18 < 19 < 20$ )

# Thuật toán song song

Độ phức tạp thuật toán:

## Định lý:

Cho một mảng  $X = (x_1, x_2, x_3 \dots x_n)$  có dạng  $x_1 < x_2 < \dots < x_n$  và một phần tử  $y$ . Thuật toán tìm kiếm song song áp dụng tìm chỉ số  $i$  sao cho  $x_i \leq y \leq x_{i+1}$  cần thời gian  $O(\log(n+1)/\log(p+1))$  với  $p$  là bộ xử lý được sử dụng.

## Chứng minh:

Trong vòng lặp: độ lớn của mảng sau mỗi lần lặp giảm dần từ  $s_i := r-1$  đến  $s_{i+1} \leq (r-1)/(p+1) + p = (s_i/p+1) + p$  đây là độ dài nhất trong  $p+1$  đoạn, đặt  $s_0 = n+1$  vậy  $s_i \leq (n+1)/(p+1)^i + p+1$  nên độ phức tạp cần thiết sẽ là  $O(\log(n+1)/\log(p+1))$ .