**Câu hỏi** 7/ Cách phân loại thứ nhất SISD (Single Instruction, Single Data) - Đơn chỉ thị, Đơn dữ liệu ?

Cách phân loại thứ nhất SISD (Single Instruction, Single Data) dùng cho máy tính đơn nguyên tuần tự

* Đơn chỉ thị: mỗi thời điểm chỉ có một chỉ thị đang được xử lý bởi CPU trong một chu kỳ đồng hồ nào đó
* Đơn luồng dữ liệu : chỉ có một dữ liệu đang được sử dụng như đầu vào trong một chu kỳ đồng hồ nào đó
* Nhận xét:
* Thực hiện theo tiền định, nghĩa là công việc được hoạch định rõ ràng khi lập trình (tĩnh), trong qúa trình thực thi các câu lệnh và dữ liệu không bị thay đổi.
* Đây là loại máy tính lâu đời nhất và thậm chí cho đến ngày nay vẫn là loại máy tính phổ biến nhất

**Câu hỏi** 8/ Cách phân loại thứ hai: SIMD (Single Instruction, Multiple Data) - Đơn chỉ thị, Đa dữ liệu ?

Cách phân loại thứ hai: SIMD (Single Instruction, Multiple Data)

* Đơn chỉ thị: Các bộ vi xử lý p1, p2, …, pn cùng thực thi một chỉ thị tại cùng một chu kỳ đồng hồ đã cho.
* Đa luồng dữ liệu: Mỗi bộ xử lý có thể hoạt động trên một phần tử dữ liệu khác với dữ liệu của bộ xử lý khác - có nhiều dữ liệu đang được sử dụng như đầu vào trong một chu kỳ đồng hồ nào đó.
* Nhận xét:
* Loại máy tính này rất phù hợp cho các vấn đề chuyên ngành được đặc trưng bởi mức đồng bộ cao, chẳng hạn như xử lý đồ họa hoặc hình ảnh,….
* Đồng bộ và thực hiện tiền định.
* Hai kiến trúc bộ xử lý hay được sử dụng nhất trong trường hợp này là: Bộ xử lý mảng và xử lý kiểu đường ống vectơ (Processor Arrays and Vector Pipelines)

**Câu hỏi** *9/* Cách phân loại thứ ba: MISD ( Multiple Instruction, Single Data)- Đa chỉ thị, Đơn dữ liệu ?

* Một dòng dữ liệu duy nhất được đưa vào các đơn vị đa xử lý.
* Mỗi đơn vị xử lý hoạt động trên các dữ liệu một cách độc lập thông qua các luồng chỉ thị độc lập khác nhau

**Câu hỏi** 10/ Cách phân loại thứ tư: MIMD ( Multiple Instruction, Multiple Data) - Đa chỉ thị, Đa dữ liệu ?

* Đa chỉ thị: Mỗi bộ xử lý có thể được thực hiện một dòng chỉ thị khác với các dòng chỉ thị của các bộ xử lý khác.
* Đa dữ liệu: Mỗi bộ xử lý có thể làm việc với một dòng dữ liệu khác với các dòng dữ liệu của các bộ xử lý khác. Việc thực hiện các chỉ thị có thể được đồng bộ hoặc không đồng bộ, tiền định hoặc không tiền định
* Nhận xét:
* Kiến trúc MIMD cũng có thể bao gồm các SIMD thực hiện các thành phần con.
* Kiến trúc MIMD còn được gọi là máy đa bộ xử lý, mỗi bộ xử lý có thể thực hiện bởi luồng lệnh và luồng dữ liệu riêng
* MIMD có bộ nhớ chung, các bộ xử lý của MIMD đều có bộ nhớ riêng, nhưng được phép truy cập vào bộ nhớ chung khi cần, do vậy giảm được các thao tác trao đổi giữa các bộ xử lý trong hệ thống

**Câu hỏi** 11/ Bộ nhớ chia sẻ truy cập ngang quyền (Uniform Memory Access (UMA)) ?

Là một bộ nhớ duy nhất được sử dụngvà truy cập bởi tất cả các bộ xử lý có hệ thống đa bộ xử lý với sự trợ giúp của mạng kết nối. Mỗi bộ xử lýcó thời gian truy cập bộ nhớ bằng nhau (độ trễ) và tốc độ truy cập.

* Ngày này hầu hết là các máy đa bộ vi xử lý đối xứng - Symmetric Multiprocessor (SMP)
* Các bộ vi xử lý giống hệt nhau
* Bình đẳng truy cập và thời gian truy cập tới bộ nhớ
* Đôi khi được gọi CC-UMA - Cache Coherent UMA. Cache coherent có nghĩa nếu một bộ vi xử lý cập nhật một vị trí trong bộ nhớ chia sẻ thì tất cả các bộ nhớ khác sẽ biết về cập nhật này. Cache coherency được thự hiện ở cấp độ phần cứng.

**Câu hỏi** 12/ Bộ nhớ chia sẻ truy cập không ngang quyền  (Non-Uniform Memory Access (NUMA)?

Là một cách [t](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B)hiết kế bộ nhớ máy tính được sử dụng trong lĩnh vực đa xử lý, trong đó thời gian truy cập bộ nhớ phụ thuộc vào vị trí tương đối của bộ nhớ so với bộ vi xử lý (CPU). Với NUMA, một bộ vi xử lý có thể truy cập bộ nhớ cục bộ của mình nhanh hơn các bộ nhớ khác (bộ nhớ của các vi xử lý khác, hoặc bộ nhớ chung chia sẻ giữa các bộ vi xử lý).

* Thường được làm bởi liên kết hai hoặc nhiều SMP
* Một SMP có thể truy cập trực tiếp bộ nhớ của một SMP khác qKhông phải tất cả các bộ xử lý có thời gia truy cập bằng nhau với tất cả bộ nhớ.
* Bộ nhớ truy cập qua liên kết thì chậm hơn
* Nếu cache coherent được duy trì thì cũng có thể được gọi là CC-NUMA - Cache Coherent NUMA

**Câu hỏi** 13/ Bộ nhớ phân tán (Distributed Memory) ?

* Giống như hệ thống bộ nhớ chia sẻ, hệ thống bộ nhớ phấn tán rất đa dạng nhưng chúng có những đặc điểm chung: Các hệ thống chia sẻ bộ nhớ yêu cầu một mạng lưới truyền thông để kết nối bộ nhớ của các bộ xử lý.
* Các bộ xử lý có bộ nhớ cục bộ riêng. Địa chỉ bộ nhớ trong một bộ xử lý không ánh xạ tới địa chỉ của bộ nhớ khác vì vậy không có khái niệm về không gian địa chỉ toàn cục trên tất cả các bộ xử lý.
* Vì mỗi bộ xử lý có bộ nhớ cục bộ riêng, nên các thao tác là độc lập. Các thay đổi được thực hiện ở bộ nhớ cục bộ mà không ảnh hưởng tới bộ nhớ trên các bộ xử lý khác. Do đó khái niệm về cache coherency không được áp dụng ở đây.
* Khi một bộ xử lý cần truy cập dữ liệu ở một bộ xử lý khác, thường là nhiệm vụ của lập trình viên cần định nghĩa tường mình cách thực hiện và khi nào dữ liệu được trao đổi. Đồng bộ hoá giữa các tác vụ là trách nhiệm của người lập trình.
* Cấu trúc mạng được sử dụng để truyền dữ liệu rất đa dạng, mặc dù vậy để đơn giản có thể dùng mạng Ethernet

**Câu hỏi** 14. Bộ nhớ hỗn hợp ?

* Các máy tính lớn nhất và nhanh nhất hiện nay đều sử dụng cả hai kiến trúc bộ nhớ này
* Thành phần bộ nhớ chia sẻ thường là một máy SMP với cache coherent. Các bộ xử lý trên SMP có thể đánh địa chỉ bộ nhớ máy tính như toàn cục.
* Thành phần trên bộ nhớ phân tán là mạng máy tính của nhiều SMP. Các SMP chỉ biết về bộ nhớ riêng của chúng, không phải bộ nhớ trên các SMP khác. Do đó các giao tiếp mạng là cần thiết để trao đổi dữ liệu từ một SMP này tới SMP khác.
* Xu hướng hiện nay dường như chỉ ra rằng kiểu kiến trúc bộ nhớ sẽ tiếp tục chiếm ưu thế và tăng khả năng tính toán trong tương lai.
* Các ưu và nhược điểm: kế thừa các ưu nhược điểm của cả hai hệ thống bộ nhớ phân tán và chia sẻ.

**Câu hỏi** 15.Các phương thức truy cập bộ nhớ ?

* **Truy cập tuần tự:** Bộ nhớ được tổ chức thành các đơn vị dữ liệu gọi là bản ghi. Việc truy cập phải được thực hiện theo một dãy tuyến tính cụ thể. Thông tin địa chỉ được lưu trữ được dùng để phân tách các bản ghi và hỗ trợ quá trình tìm kiếm lấy thông tin. Một bộ phận đọc/ghi dùng chung được sử dụng. Bộ phận này phải được di chuyển từ vị trí hiện thời của nó đến vị trí được yêu cầu, quét qua và từ chối các bản ghi trung gian. Do đó, thời gian để truy cập một bản ghi tùy ý biến đổi khá cao. Các đơn vị băng từ, được thảo luận trong chương 5, là các đơn vị có dạng truy cập tuần tự.
* **Truy cập trực tiếp:** Cũng như với truy cập tuần tự, truy cập trực tiếp bao gồm việc dùng chung một bộ phận đọc/ghi. Tuy nhiên, các khối hay bản ghi riêng lẻ có một địa chỉ duy nhất dựa trên vị trí vật lý. Việc truy cập được thực hiện thông qua truy cập trực tiếp cộng với tìm kiếm tuần tự, đếm, hay chờ để đến được vị trí cuối cùng. Một lần nữa, thời gian truy cập là biến đổi. Các đơn vị đĩa được trình bày trong chương 5 là các đơn vị truy cập trực tiếp.
* **Truy cập ngẫu nhiên:**Mỗi vị trí khả định địa chỉ trong bộ nhớ có motä cơ chế định địa chỉ vật lý duy nhất. Thời gian truy cập một vị trí cho trước độc lập với dãy các truy cập trước đó và không thay đổi. Do đó, bất kỳ một vị trí nào cũng có thể được chọn ngẫu nhiên và được định địa chỉ cũng như truy cập trực tiếp. Các hệ thống bộ nhớ chính được truy cập ngẫu nhiên.
* **Liên kết:**Đây là kiểu truy cập ngẫu nhiên bộ nhớ cho phép thực hiện việc so sánh các vị trí bit có yêu cầu trong một word phục vụ cho việc đối sánh đặc biệt nào đó, và có thể thực hiện thao tác này cùng một lúc cho tất cả các word. Do đó một word được trích ra dựa trên một phần nội dung của nó chứ không phải dựa trên địa chỉ. Tương tự như với phương thức truy cập ngẫu nhiên thông thường, mỗi vị trí nhớ có cơ chế định địa chỉ riêng, và thời gian lấy thông tin không đổi, độc lập với vị trí hoặc khuôn dạng truy cập trước đó. Bộ nhớ cache, được đề cập đến trong mục 4.3 có thể tận dụng cách truy cập liên kết này.