Linked List:

**Stack**:

Java Virtual Machine uses a stack

The Java compiler translates programs into posfix notation

**Queue**:

Máy in: hệ điều hành sử dụng '' hàng đợi in '' để giữ danh tính của các tiến trình (thực thi các chương trình người dùng) muốn sử dụng máy in. Khi một quy trình bắt đầu sử dụng máy in, nó được phép tiếp tục cho đến khi hoàn tất. Các quy trình khác muốn sử dụng máy in phải đợi và ID của chúng (và địa chỉ của tệp cần in) được giữ trong hàng đợi. Khi quá trình sử dụng máy in kết thúc, quá trình được giữ lâu nhất trong hàng đợi sẽ được truy xuất và lệnh in của nó được bắt đầu.

Ổ đĩa trong ('' cứng ''): một đĩa có thể thực hiện nhiều nhất một lần đọc hoặc ghi bất kỳ lúc nào, vì vậy Hệ điều hành sử dụng hàng đợi để lưu giữ các yêu cầu của quy trình (ID của quy trình và thông tin chi tiết về việc đọc / ghi) để đọc hoặc ghi từ một sector trên đĩa. Bộ điều khiển của đĩa lấy lại từ hàng đợi yêu cầu tiếp theo '' trong dòng '' và thực hiện nó. (Khi không có yêu cầu nào nữa, bộ điều khiển '' ngủ ''.)

Hệ điều hành thường duy trì một hàng đợi các quá trình đã sẵn sàng thực thi hoặc hàng đợi một sự kiện cụ thể xảy ra.

Hệ thống máy tính thường phải cung cấp một "vùng lưu giữ" cho các thông báo giữa hai quy trình hoặc chương trình nội bộ, hoặc giữa hai hệ thống qua mạng. Vùng lưu giữ này thường được gọi là "vùng đệm" và thường được thực hiện như một hàng đợi, vì chúng ta muốn thứ tự thời gian thông báo được giữ lại.

Hệ thống máy tính thường phải cung cấp một "vùng lưu giữ" cho các thông báo giữa hai quy trình hoặc chương trình nội bộ, hoặc giữa hai hệ thống qua mạng. Vùng lưu giữ này thường được gọi là "vùng đệm" và thường được thực hiện như một hàng đợi, vì chúng ta muốn thứ tự thời gian thông báo được giữ lại.

Hệ thống điện thoại Call Center sử dụng Hàng đợi để giữ những người gọi cho họ theo thứ tự, cho đến khi đại diện dịch vụ miễn phí

Xử lý ngắt trong hệ thống thời gian thực. Các ngắt được xử lý theo thứ tự khi chúng đến, tức là Ai đến trước được phục vụ trước

**Hash Table:**

Dict or dictionary in Python, or HashMap in Java

Các từ khóa của một ngôn ngữ lập trình được xử lý khác với các định danh khác. Để phân biệt giữa các từ khóa của ngôn ngữ lập trình (if, else, for, return, v.v.) và các số nhận dạng khác và để biên dịch thành công chương trình, trình biên dịch lưu trữ tất cả các từ khóa này trong một tập hợp được triển khai bằng bảng băm

Một trong những ứng dụng nổi tiếng nhất của băm là thuật toán Rabin-Karp. Về cơ bản, đây là một thuật toán tìm kiếm chuỗi sử dụng phép băm để tìm bất kỳ một bộ mẫu nào trong một chuỗi. Một ứng dụng thực tế của thuật toán này là phát hiện đạo văn. Để biết thêm về Rabin-Karp algo

Khi di chuyển qua các tệp trên hệ thống cục bộ, hai thành phần rất quan trọng của tệp, đó là tên tệp và đường dẫn tệp. Để lưu trữ sự tương ứng giữa tên tệp và đường dẫn tệp, hệ thống sử dụng một bản đồ (tên tệp, đường dẫn tệp) được triển khai bằng bảng băm

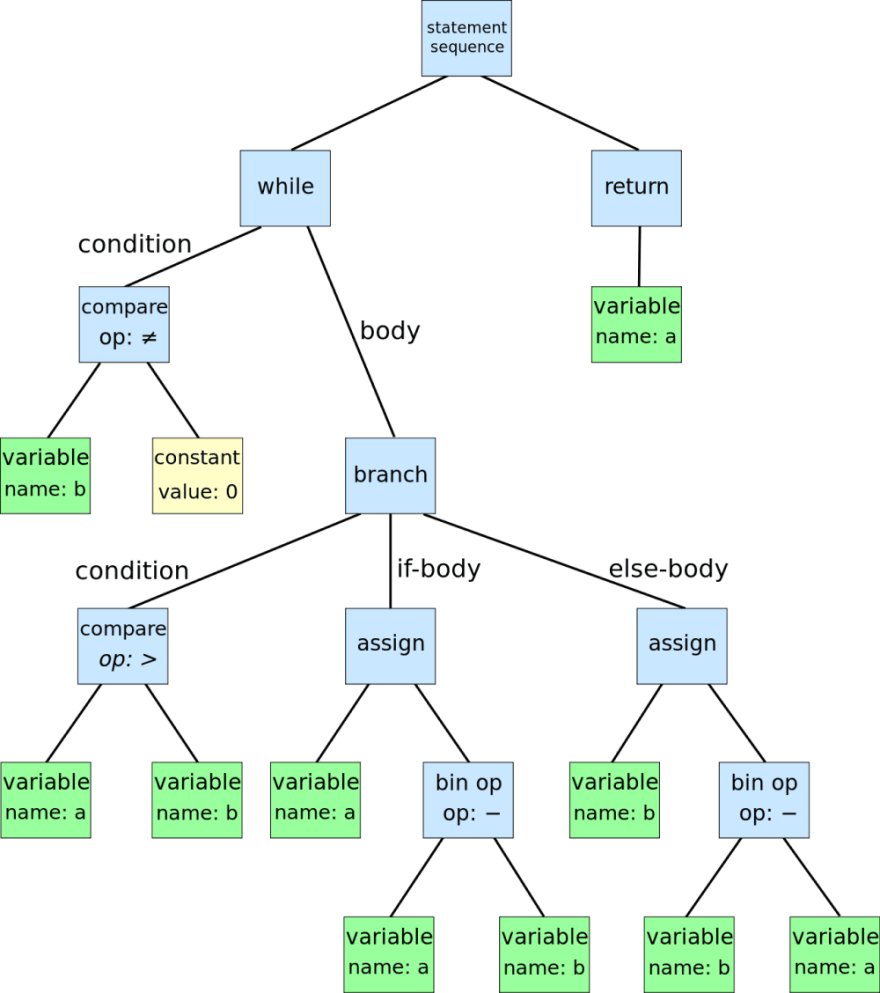
**Binary Search Tree:**

Được sử dụng trong nhiều ứng dụng tìm kiếm nơi dữ liệu liên tục vào / ra, chẳng hạn như map và set trong các thư viện của nhiều ngôn ngữ lập trình

Thường được sử dụng trong cơ sở dữ liệu và hệ thống tệp để lưu trữ và truy xuất dữ liệu một cách hiệu quả

Hầu hết mọi công cụ 3D đều sử dụng BST để xác định những đối tượng nào nên được hiển thị trong thế giới 3D, về cơ bản là sắp xếp theo thứ tự Z thay vì tính toán lại vị trí 3D phức tạp trên mỗi lần kết xuất

Để phân tích cú pháp hiệu quả các biểu thức cú pháp (ví dụ: ngôn ngữ lập trình), trình biên dịch sử dụng một BST đặc biệt được gọi là cây cú pháp.



Tổ chức của [mã Morse](https://en.wikipedia.org/wiki/Morse_code) là một cây nhị phân.

Nguồn: <http://people.cs.ksu.edu/~schmidt/300s05/Lectures/Week3.html>