**TRẢ LỜI CÂU HỎI – CHƯƠNG 2**

**Họ và tên:** Nguyễn Thị Yến Khương

**MSSV:** 1654050052 – **Nhóm: 04**

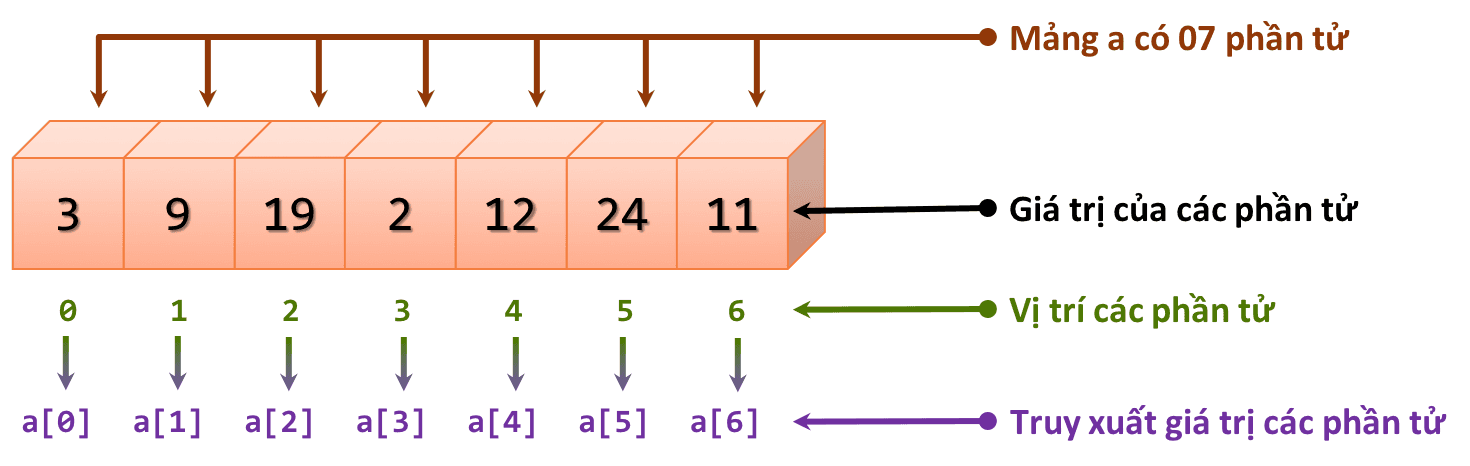
**Bài làm:**

**Câu 1: Trong Khoa học máy tính, Danh sách đặc được hiểu như thế nào? Cho ví dụ.**

Trong Khoa học máy tính, Danh sách đặc hay còn được biết đến với tên gọi tiếng anh là List, là một danh sách mà các phần tử trong danh sách có **cùng kiểu dữ liệu** và được **cấp phát liên tục trong bộ nhớ**. Trong Danh sách đặc, tổng số lượng phần tử tối đa phải được biết trước và cố định, mỗi phần tử của danh sách thường chứa một giá trị và ở một vị trí không đổi đã được định nghĩa trước đó.

Ngoài ra, Danh sách đặc còn được đánh giá là một cấu trúc dữ liệu “đẹp” bởi tính đơn giản của nó. Chính vì thế, mà Danh sách đặc đặc biệt thích hợp cho các bài toán mà ta đã biết trước được số lượng phần tử, hoặc có thể xác định được khi chạy chương trình.

Ví dụ Danh sách đặc:



Cho một danh sách đặc có 7 phần tử, với 0, 1, 2, 3,…, 6 là chỉ số từng phần tử trong danh sách.

a[0] là biến chứa chứa giá trị của danh sách tại vùng có chỉ số 0 (a[0] = 3).

a[1] là biến chứa chứa giá trị của danh sách tại vùng có chỉ số 1 (a[1] = 9).

…

a[6] là biến chứa chứa giá trị của danh sách tại vùng có chỉ số 6 (a[6] = 11).

**Câu 2: Trong Khoa học máy tính, Danh sách liên kết được hiểu như thế nào? Có mấy loại? Cho ví dụ.**

Trong Khoa học máy tính, Danh sách liên kết, hay còn được biết đến với tên gọi tiếng anh là Linked List, là một dãy các cấu trúc dữ liệu được kết nối với nhau thông qua các liên kết (link). Hiểu một cách đơn giản thì Danh sách liên kết là một cấu trúc dữ liệu bao gồm một nhóm các nút (node) tạo thành một chuỗi. Trong đó, các phần tử này được cấp phát rời rạc nhau và cố định trong bộ nhớ. Mỗi phần tử trong danh sách liên kết gồm vùng thông tin chứa dữ liệu cần quản lý và vùng liên kết dùng để tham chiếu đến bộ nhớ của nút kế tiếp trong chuỗi.

Phân loại Danh sách liên kết và cho ví dụ:

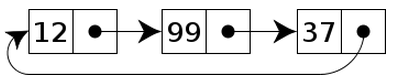
* Danh sách liên kết đơn (Simple Linked List): chỉ duyệt các phần tử theo chiều về trước.



* Danh sách liên kết đôi (Double Linked List): các phần tử có thể được duyệt theo chiều về trước hoặc về sau.



* Danh sách liên kết vòng (Circular Linked List): phần tử cuối cùng chứa link của phần tử đầu tiên.



**Câu 3: Tại sao nói STACK và QUEUE là danh sách hạn chế? Cho ví dụ.**

**Stack** hay còn gọi là ngăn xếp, là một tập hợp các phần tử có cùng kiểu dữ liệu được tổ chức một cách tuần tự. Trong đó, các phần tử được thêm vào và lấy ra chỉ ở một đầu của danh sách. Chính vì thế, việc thêm/lấy một đối tượng ra khỏi Stack đều được thực hiện theo cơ chế LIFO (Last in – First out). Đặc biệt, các phần tử của Stack có thể được thêm vào bất kỳ lúc nào, nhưng chỉ phần tử được thêm vào sau cùng mới được truy xuất đầu tiên. Trong thuật ngữ Stack, thao tác thêm một đối tượng vào được gọi là “Push” và lấy đối tượng ra được gọi là “Pop”.

**Queue** hay còn gọi là hàng đợi, là một tập hợp các phần tử cùng kiểu dữ liệu được tổ chức một cách tuần tự. Trong đó, việc thêm một phần tử vào Queue sẽ được diễn ra ở cuối Queue và việc lấy một phần tử ra khỏi Queue luôn diễn ra ở đầu Queue. Vì thế, cấu trúc hoạt động của Queue dựa theo cơ chế FIFO (First in – First out). Hiểu một cách đơn giản, phần tử được thêm vào trước, sẽ được lấy/xóa ra trước.

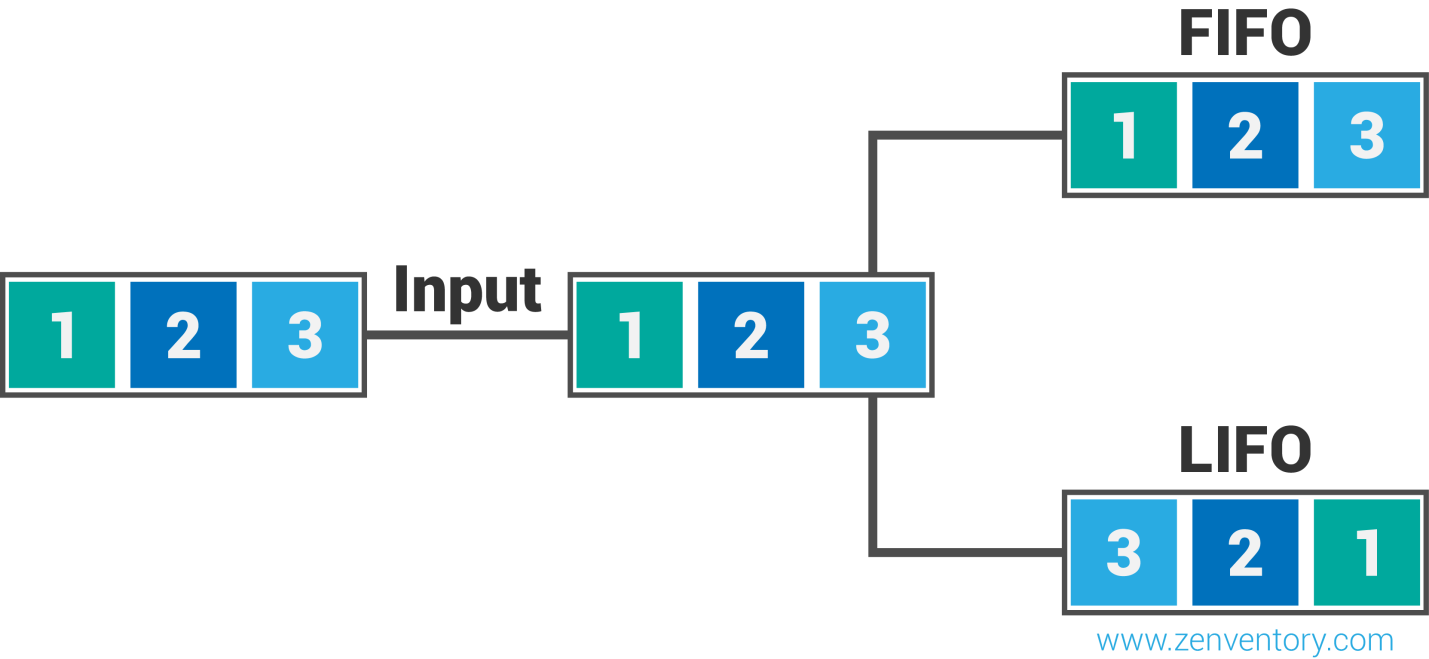
**Stack** và **Queue** còn được gọi là danh sách hạn chế bởi vì sự hạn chế bởi các theo tác truy cập tới các phần tử của danh sách đó. Đối với Stack, khi muốn thao tác trên dữ liệu thì phải bắt buộc tuân theo cơ chế LIFO, ta giới hạn việc thêm vào hoặc loại bỏ một phần tử chỉ thực hiện tại một đầu của danh sách, đầu này gọi **là** đỉnh (TOP) của ngăn xếp. Còn với Queue thì theo cơ chế FIFO, cho phép việc thêm vào một phần tử chỉ thực hiện tại một đầu của danh sách, còn đầu kia được sử dụng để xóa dữ liệu loại bỏ phần tử. Không như Danh sách đặc hay Danh sách liên kết đơn, ta có thể thêm/xóa một phần tử ở một vị trí bất kỳ trong danh sách, nhưng còn đối với Stack và Queue thì ta không thể thực hiện truy xuất ngẫu nhiên như vậy được, mà phải theo một cơ chế cố định tùy theo từng loại. Đó là lí do vì sao mà Stack và Queue đều được xếp vào nhóm Danh sách hạn chế.

**Câu 4: Thế nào là LIFO, FIFO?**

**FIFO** là từ viết tắt cho “First in – First out”, có nghĩa là vào trước thì ra trước. Với phương pháp này, các phần tử đầu tiên được nhập vào danh sách sẽ là phần tử đầu tiên ra khỏi danh sách.

**LIFO** là từ viết tắt cho “Last in – First out”, có nghĩa là vào sau thì ra trước. Với phương pháp này, các phần tử được nhập vào danh sách gần nhất sẽ là phần tử ra phải danh sách đầu tiên.

Ví dụ về LIFO và FIFO: Giả sử có 3 phần tử với giá trị là 1, 2, 3 và được nhập vào danh sách theo thứ tự như hình bên dưới (3 vào trước, sau đó tới 2, và cuối cùng là 1). Đối với cơ chế FIFO (First in – First out), phần tử 3 vào trước sẽ ra trước, sau đó tới 2, cuối cùng là 1. Ngược với FIFO, cơ chế LIFO (Last in – First out) được thực hiện như sau, 1 là phần tử được nhập vào danh sách gần nhất, tức được nhập cuối cùng, thì 1 sẽ là phần tử ra khỏi danh sách đầu tiên, tiếp theo là 2 và cuối cùng là 3.



**Câu 5: Theo bạn, danh sách liên kết có thể ứng dụng xử lý các vấn đề gì trong máy tính?**

Danh sách liên kết có nhiều ứng dụng nổi bật nhằm xử lý các vấn đề trong máy tính, đặc biệt là ứng dụng trong Hệ điều hành. Trong Hệ điều hành, Danh sách liên kết được ứng dụng phổ biến cho việc hiện thực hệ thống File. Đây là một trong những phương pháp dùng để cấp phát/tổ chức không gian cho File dựa trên cơ chế cấp phát theo Danh sách liên kết.

Cấp phát không gian cho File dùng Danh sách liên kết: Trong Hệ điều hành, mỗi tập tin lưu nội dung trên một block (khối lưu trữ) của thiết bị lưu trữ.

* Mỗi tập tin chiếm một tập các block được kết nối với nhau tạo thành một Danh sách liên kết, và các block này có thể nằm rãi rác trên đĩa. Trong đó, mỗi block sẽ chứa dữ liệu của tập tin và con trỏ trỏ đến địa chỉ của block kế tiếp của tập tin (≈ 4 bytes).
* Có một Bảng thư mục (Directory), mỗi mục trong Bảng thư mục chứa số hiệu của block đầu tiên (start) và số hiệu của block kết thúc (end).
* Ưu điểm của cấp phát theo kiểu Danh sách liên kết: Đơn giản, chỉ cần quản lý vị trí (số hiệu) của block bắt đầu; Không cần biết trước kích thước của tập tin; Không xảy ra hiện tượng phân mảnh ngoại vi;…

**Câu 6: Thế nào là Cấu trúc dữ liệu động? Cho ví dụ.**

Trong nhiều trường hợp, tại một thời điểm biên dịch không thể xác định trước kích thước chính xác của một số đối tượng dữ liệu do sự tồn tại và tăng trưởng của chúng phụ thuộc vào ngữ cảnh của việc thực hiện chương trình. Do đó, Cấu trúc dữ liệu động được định nghĩa là cấu trúc dữ liệu chứa một số đối tượng dữ liệu có thể thay đổi được kích thước, cấu trúc,… trong suốt quá trình sống. Các đối tượng dữ liệu thuộc những kiểu dữ liệu này gọi là kiểu dữ liệu động. Ngoài ra, Cấu trúc dữ liệu động sẽ được cấp phát động trong lúc chạy chương trình, với các phần tử của cấu trúc có thể nằm rải rác ở nhiều nơi và được cố định trong bộ nhớ.

Ví dụ Cấu trúc dữ liệu động: Điển hình của Cấu trúc dữ liệu động là Danh sách liên kết (gồm Danh sách liên kết đơn và Danh sách liên kết đôi) với độ linh hoạt cao, có thể thay đổi kích thước và câu trúc dễ dàng.