**4.2. BIỂU DIỄN DANH SÁCH TRONG MÁY TÍNH**

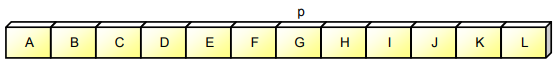
Việc cài đặt một danh sách trong máy tính tức là tìm một cấu trúc dữ liệu cụ thể mà máy tính hiểu được để lưu các phần tử của danh sách đồng thời viết các đoạn chương trình con mô tả các thao tác cần thiết đối với danh sách.

**4.2.1. Cài đặt bằng mảng một chiều**

Khi cài đặt danh sách bằng một mảng, thì có một biến nguyên n lưu số phần tử hiện có trong danh sách. Nếu mảng được đánh số bắt đầu từ 1 thì các phần tử trong danh sách được cất giữ trong mảng bằng các phần tử được đánh số từ 1 tới n.

**Chèn phần tử vào mảng**:

Mảng ban đầu:



Nếu muốn chèn một phần tử V vào mảng tại vị trí p, ta phải:

Dồn tất cả các phần tử từ vị trí p tới tới vị trí n về sau một vị trí:



Đặt giá trị V vào vị trí p:



Tăng n lên 1

**Xoá phần tử khỏi mảng**:

Mảng ban đầu:



Muốn xoá phần tử thứ p của mảng mà vẫn giữ nguyên thứ tự các phần tử còn lại, ta phải: Dồn tất cả các phần tử từ vị trí p + 1 tới vị trí n lên trước một vị trí:



Giảm n đi 1



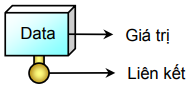
Trong trường hợp cần xóa một phần tử mà không cần duy trì thứ tự của các phần tử khác, ta chỉ cần đảo giá trị của phần tử cần xóa cho phần tử cuối cùng rồi giảm số phần tử của mảng (n) đi 1.

**4.2.2. Cài đặt bằng danh sách liên kết đơn**

Danh sách liên kết đơn gồm các nút được nối với nhau theo một chiều. Mỗi nút là một bản ghi (record) gồm hai trường:

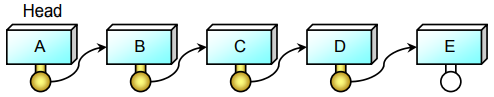
Trường thứ nhất chứa giá trị lưu trong nút đó

Trường thứ hai chứa liên kết (con trỏ) tới nút kế tiếp, tức là chứa một thông tin đủ để biết nút kế tiếp nút đó trong danh sách là nút nào, trong trường hợp là nút cuối cùng (không có nút kế tiếp), trường liên kết này được gán một giá trị đặc biệt.



**Cấu trúc nút của danh sách liên kết đơn**

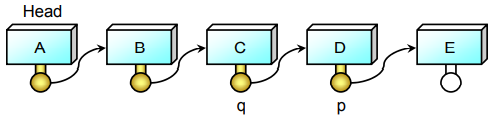
Nút đầu tiên trong danh sách được gọi là chốt của danh sách liên kết đơn (Head). Để duyệt danh sách liên kết đơn, ta bắt đầu từ chốt, dựa vào trường liên kết để đi sang nút kế tiếp, đến khi gặp giá trị đặc biệt (duyệt qua nút cuối) thì dừng lại



**Danh sách liên kết đơn**

**Chèn phần tử vào danh sách liên kết đơn**:

Danh sách ban đầu:



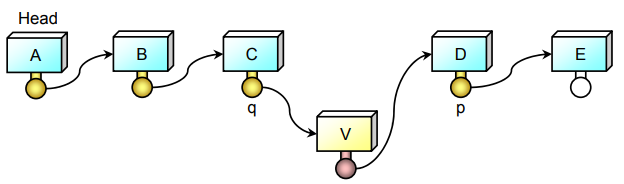
Muốn chèn thêm một nút chứa giá trị V vào vị trí của nút p, ta phải:

a) Tạo ra một nút mới NewNode chứa giá trị V:



b) Tìm nút q là nút đứng trước nút p trong danh sách (nút có liên kết tới p).

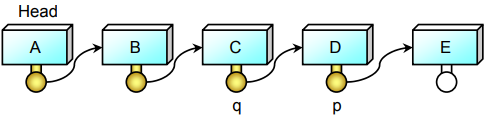
b1) Nếu tìm thấy thì chỉnh lại liên kết: q liên kết tới NewNode, NewNode liên kết tới p



b2) Nếu không có nút đứng trước nút p trong danh sách thì tức là p = Head, ta chỉnh lại liên kết: NewNode liên kết tới Head (cũ) và đặt lại Head = NewNode

**Xoá phần tử khỏi danh sách liên kết đơn**:

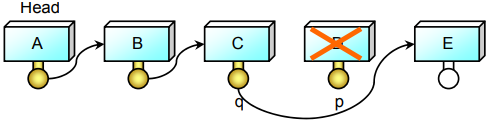
Danh sách ban đầu:



Muốn huỷ nút p khỏi danh sách liên kết đơn, ta phải:

Tìm nút q là nút đứng liền trước nút p trong danh sách (nút có liên kết tới p)

Nếu tìm thấy thì chỉnh lại liên kết: q liên kết thẳng tới nút liền sau p, khi đó quá trình duyệt danh sách bắt đầu từ Head khi duyệt tới q sẽ nhảy qua không duyệt p nữa. Trên thực tế khi cài đặt bằng các biến động và con trỏ, ta nên có thao tác giải phóng bộ nhớ đã cấp cho nút p

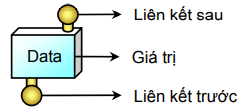


Nếu không có nút đứng trước nút p trong danh sách thì tức là p = Head, ta chỉ việc đặt lại Head bằng nút đứng kế tiếp Head (cũ) trong danh sách. Sau đó có thể giải phóng bộ nhớ cấp cho nút p (Head cũ)

**4.2.3. Cài đặt bằng danh sách liên kết đôi**

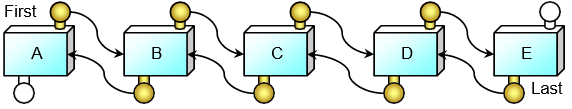
Danh sách liên kết đôi gồm các nút được nối với nhau theo hai chiều. Mỗi nút là một bản ghi (record) gồm ba trường:48

* Trường thứ nhất chứa giá trị lưu trong nút đó
* Trường thứ hai (Next) chứa liên kết (con trỏ) tới nút kế tiếp, tức là chứa một thông tin đủ để biết nút kế tiếp nút đó là nút nào, trong trường hợp là nút cuối cùng (không có nút kế tiếp), trường liên kết này được gán một giá tị đặc biệt.
* Trường thứ ba (Prev) chứa liên kết (con trỏ) tới nút liền trước, tức là chứa một thông tin đủ để biết nút đứng trước nút đó trong danh sách là nút nào, trong trường hợp là nút đầu tiên (không có nút liền trước) trường này được gán một giá trị đặc biệt.



**Cấu trúc nút của danh sách liên kết đôi**

Khác với danh sách liên kết đơn, danh sách liên kết đôi có hai chốt: Nút đầu tiên trong danh sách được gọi là First, nút cuối cùng trong danh sách được gọi là Last. Để duyệt danh sách liên kết đôi, ta có hai cách: Hoặc bắt đầu từ First, dựa vào liên kết Next để đi sang nút kế tiếp, đến khi gặp giá trị đặc biệt (duyệt qua nút cuối) thì dừng lại. Hoặc bắt đầu từ Last, dựa vào liên kết Prev để đi sang nút liền trước, đến khi gặp giá trị đặc biệt (duyệt qua nút đầu) thì dừng lại

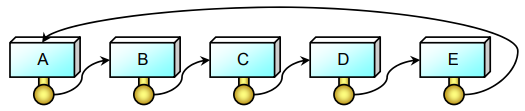


**Danh sách liên kết đôi**

Việc chèn / xoá vào danh sách liên kết đôi cũng đơn giản chỉ là kỹ thuật chỉnh lại các mối liên kết giữa các nút cho hợp lý, ta coi như bài tập.

**4.2.4. Cài đặt bằng danh sách liên kết vòng một hướng**

Trong danh sách liên kết đơn, phần tử cuối cùng trong danh sách có trường liên kết được gán một giá trị đặc biệt (thường sử dụng nhất là giá trị nil). Nếu ta cho trường liên kết của phần tử cuối cùng trỏ thẳng về phần tử đầu tiên của danh sách thì ta sẽ được một kiểu danh sách mới gọi là danh sách liên kết vòng một hướng.

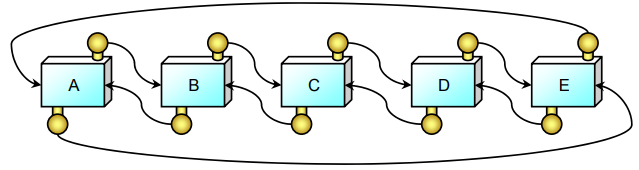


**Danh sách liên kết vòng một hướng**

Đối với danh sách liên kết vòng, ta chỉ cần biết một nút bất kỳ của danh sách là ta có thể duyệt được hết các nút trong danh sách bằng cách đi theo hướng của các liên kết. Chính vì lý do này, khi chèn xoá vào danh sách liên kết vòng, ta không phải xử lý các trường hợp riêng khi chèn xoá tại vị trí của chốt

**4.2.5. Cài đặt bằng danh sách liên kết vòng hai hướng**

Danh sách liên kết vòng một hướng chỉ cho ta duyệt các nút của danh sách theo một chiều, nếu cài đặt bằng danh sách liên kết vòng hai hướng thì ta có thể duyệt các nút của danh sách cả theo chiều ngược lại nữa. Danh sách liên kết vòng hai hướng có thể tạo thành từ danh sách liên kết đôi nếu ta cho trường Prev của nút First trỏ thẳng tới nút Last còn trường Next của nút Last thì trỏ thẳng về nút First.



**Danh sách liên kết vòng hai hướng**