1. Khái niệm mô hình dữ liệu, phân loại? VD?

- khái niệm Mô hình dữ liệu là mô hình toán học cùng với các phép toán có thể thực hiện trên các đối tượng của mô hình. Mô hình dữ liệu được sử dụng để mô tả cấu trúc

logic của dữ liệu được xử lý bởi hệ thống.

- Phân loại :

+ Mô hình dữ liệu tuyến tính (danh sách) : dùng để biểu diễn các phần tử có quan hệ 1:1

+ Mô hình dữ liệu phân cấp (mô hình cây) : dùng để biểu diễn các phần tử có quan hệ 1:n

+ Mô hình dữ liệu đồ thị : đây là mô hình dữ liệu phong phú và phức tạp. Các phần tử có mối quan hệ n:m

+ Mô hình dữ liệu tập hợp : trong 1 tập hợp, các phần tử không có mỗi quan hệ trực tiếp với nhau, giữa chúng chủ có mối quan hệ là thành viên của 1 tập hợp

Ví dụ: Mô hình đồ thị, cây, danh sách, tập hợp, ..

1. Khái niệm kiểu dữ liệu trừu tượng? VD?

-KN : là một mô hình dữ liệu được xét cùng với một số xác định các phép toán.

+ Trong Mô hình dữ liệu, chúng ta có thể thực hiện một tập hợp các phép toán rất đa dạng, phong phú. Song trong nhiều áp dụng, chúng ta chỉ sử dụng mô hình với một số xác định các phép toán nào đó. Khi đó chúng ta sẽ có một kiểu dữ liệu trừu tượng

- Ví dụ: mô hình dữ liệu danh sách, chỉ xét đến các phép toán thêm vào và lấy ra, ta gọi là KDLTT hàng đợi hoặc KDLTT ngăn xếp,

1. Mối quan hệ giữa CTDL&TT

- Trong một chương trình máy tính, giải thuật và cấu trúc dữ liệu có mối quan hệ chặt chẽ với nhau, được thể hiện qua công thức

Cấu trúc dữ liệu + Giải thuật = chương trình.

-Với một cấu trúc dữ liệu đã chọn, sẽ có những giải thuật tương ứng, phù hợp.

-Khi cấu trúc dữ liệu thay đổi thường giải thuật cũng phải thay đổi theo để tránh việc xử lý gượng ép, thiếu tự nhiên trên một cấu trúc không phù hợp.

-Hơn nữa, một cấu trúc dữ liệu tốt sẽ giúp giải thuật xử lý trên đó có thể phát huy tác dụng tốt hơn, vừa đáp ứng nhanh vừa tiết kiệm bộ nhớ, giải thuật cũng dễ hiễu và đơn giản hơn.

1. Vai trò của CTDL? Các tiêu chí lựa chọn CTDL?

Cấu trúc dữ liệu  
Cấu trúc dữ liệu được mô tả như sau:  
CTDL = { Phần tử dữ liệu } (1)  
Trong đó, các dữ liệu thành phần thuộc kiểu liệu đã được định nghĩa trước, chúng có một mối liên kết nào đó với nhau. Một cách khái quát, cấu trúc dữ liệu là một dạng biểu diễn của các dữ liệu, mối quan hệ và các thao tác có thể tác động lên các phần tử dữ liệu này.

Ví dụ: cấu trúc dữ liệu mảng bao gồm một dãy có thứ tự các phần tử có cùng 1 kiểu dữ liệu xác định nào đó và các thao tác cơ bản có thể thực hiện trên các phần tử của mảng; bản ghi: bao gồm một tập các phần tử có thể thuộc các kiểu dữ liệu khác nhau, có mối quan hệ với nhau đó là chúng cùng mô tả các thuộc tính của một đối tượng, ví dụ thông tin về 1 con người gồm họ tên, ngày sinh, chiều cao, cân nặng, ....và các thao tác cơ bản có thể thực hiện trên các thành phần của bản ghi.

Các tiêu chí lựa chọn CTDL?

- Phản ánh đúng thực tế : Đây là tiêu chuẩn quan trọng nhất, quyết định tính đúng đắn của toàn bộ bài toán.

- Phù hợp với các thao tác trên đó : Tiêu chuẩn này giúp tăng tính hiệu quả của giải thuật, cụ thể là giúp cho việc phát triển các thuật toán đơn giản, tự nhiên hơn trên cấu trúc.

- Tiết kiệm tài nguyên hệ thống : Cấu trúc dữ liệu chỉ nên sử dụng tài nguyên hệ thống vừa đủ để đảm nhiệm được chức năng của nó.

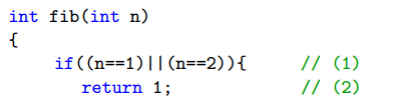
1. Khái niệm đệ quy, giải thuật đệ quy? VD?

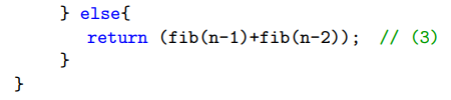
- KN: là một kĩ thuật dùng để định nghĩa một khái niệm trực tiếp hoặc gián tiếp theo chính nó. Một đối tượng được gọi là đệ quy nếu nó bao gồm chính nó như một bộ phận hoặc nó được định nghĩa dưới dạng của chính nó.

- Định nghĩa giải thuật đệ quy : Nếu lời giải của bài toán T được thực hiện bởi lời giải của bài toán T’ có dạng như T thì đó là một lời giải đệ quy

- Giải thuật chứa lời giải đệ quy gọi là giải thuật đệ quy (T’< T)

VD: Chương trình con đệ quy là chương trình con trong đó có chứa lời gọi tới chính nó. Ví dụ về chương trình con đệ quy cho bài toán tìm số Fibonacci thứ n:





1. Các đặc điểm của giải thuật đệ quy? Viết giải thuật đệ quy tính n!

ĐẶC ĐIỂM CỦA GIẢI THUẬT ĐỆ QUY

- (1) Trong giải thuật đệ quy bao giờ cũng có lời gọi đến chính tên giải thuật.

- (2) Mỗi lần có lời gọi thì kích thước của bài toán thu nhỏ hơn trước

- (3) Có một trường hợp đặc biệt, trường hợp suy biến: Bài toán sẽ được giải quyết theo một cách khác hẳn và gọi đệ quy cũng kết thúc.

Có 2 loại đệ quy, đệ quy trực tiếp (thủ tục chứa lời gọi đến chính nó) và đệ quy gián tiếp (thủ tục chứa lời gọi đến thủ tục khác mà thủ tục này lại chứa lời gọi đến chính nó)

GIẢI THUẬT ĐỆ QUY TÍNH n!

Xét hàm đệ quy tính n!

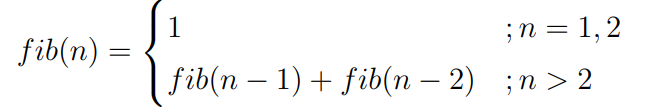
int fact(int n)  
{  
if(n <=1){ // (1)  
return 1; // (2)  
} else{  
return n\*fact(n-1); // (3)  
}  
}

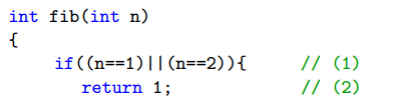
1. Công thức tính số Fibonacci

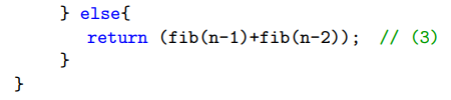
Nếu n=1, hoặc n=2 thì F(n)=1 ( phần đk dừng).

Nếu n>2 thì F(n)= F(n-1)+F(n-2) ( phần đệ quy)

Cho Fn-1 và Fn-2 , tính Fn







1. Khái niệm danh sách , ví dụ?

KN: Mô hình toán học của danh sách là một tập hợp hữu hạn biến động các phần tử thuộc cùng một lớp đối tượng nào đó (có cùng một kiểu dữ liệu). Ta cũng lưu ý rằng một đối tượng cũng có thể xuất hiện nhiều lần trong một danh sách. Ta biểu diễn danh sách L

như là một chuỗi các phần tử của nó: a1, a2, . . ., an với n ≥ 0. thì:

+ Nếu n = 0 ta nói *danh sách rỗng* (empty list).

+ Nếu n > 0 ta gọi a1 là phần tử đầu tiên và an là phần tử cuối cùng của danh sách.

+ Số phần tử của danh sách ta gọi là độ dài của danh sách.

+ Một tính chất quan trọng của danh sách đó là tính tuyến tính: Các phần tử của danh sách có thứ tự tuyến tính theo vị trí (position) xuất hiện của các phần tử. Ta nói ai đứng trước ai+1, với i từ 1 đến n-1; Tương tự ta nói ai là phần tử đứng sau ai-1, với i từ 2 đến n.

VD: Tập hợp họ tên các sinh viên của lớp TINHOC được liệt kê trên giấy như sau:

1. Nguyễn Trung Cang

2. Nguyễn Ngọc Chương

3. Lê Thị Lệ Sương

4. Trịnh Vũ Thành

5. Nguyễn Phú Vĩnh

6. Phạm Quách Què

• Là một danh sách gồm có 6 phần tử, các pt có kiểu dữ

liệu xâu ký tự, mỗi phần tử có một vị trí trong danh

sách theo thứ tự xuất hiện của nó, có quan hệ tuyến

tính với các phần tử đứng trước, đứng sau.

1. So sánh danh sách kế tiếp và dánh sách liên kết

|  |  |
| --- | --- |
| ****Danh sách kế tiếp (Array/List):**** | ****Danh sách liên kết (Linked List):**** |
| * Là một cấu trúc dữ liệu trong đó mỗi phần tử (node) chứa dữ liệu và một liên kết đến phần tử tiếp theo trong danh sách. * Không yêu cầu một phần tử có vùng nhớ liền trước đó trong bộ nhớ. Các phần tử có thể nằm rải rác trong bộ nhớ. * Có thể thêm hoặc xóa một phần tử mà không cần phải di chuyển các phần tử khác.   -có thể được duyệt bằng chỉ mục, nhanh chóng  -có thể làm nhanh chóng nếu chỉ cần thêm/xóa ở cuối danh sách  -có thể sử dụng một lượng bộ nhớ linh hoạt hơn. | * Là một cấu trúc dữ liệu trong đó mỗi phần tử được đánh số và có thể truy cập trực tiếp thông qua chỉ mục. * Yêu cầu một vùng nhớ liên tục để lưu trữ các phần tử. * Có thể thao tác nhanh với chỉ mục, nhưng việc thêm/xóa phần tử giữa danh sách có thể đòi hỏi di chuyển một số phần tử.   -cần phải đi qua từ đầu đến phần tử mong muốn, có thể mất thời gian hơn.  -thường nhanh hơn nếu cần thêm/xóa ở giữa danh sách.  -yêu cầu một lượng bộ nhớ liên tục. |

1. Khái niệm ngăn xếp hàng đợi , ví dụ?

KN ngăn xếp : Ngăn xếp (Stack) là một danh sách đặc biệt, trong đó việc thêm vào, lấy một phần tử ra khỏi ngăn xếp chỉ thực hiện tại một đầu của ngăn xếp, đầu này gọi là đỉnh (TOP) của ngăn xếp.

Ví dụ: Xếp chồng đĩa CD

KN hàng đợi: Hàng đợi, hay ngắn gọn là hàng (queue) là một danh sách đặc biệt mà phép thêm một phần tử vào hàng chỉ thực hiện tại một đầu của hàng, gọi là cuối hàng (REAR), còn phép lấy một phần tử ra khỏi hàng thì thực hiện ở đầu còn lại của hàng, gọi là đầu hàng (FRONT).

Ví dụ: Khi ta xếp hàng mua vé xem phim

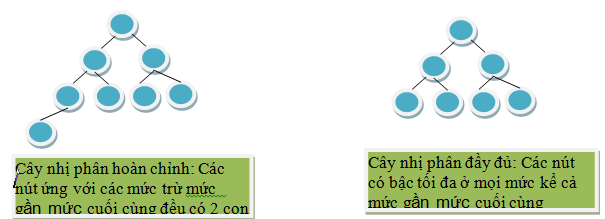
1. So sánh ngăn xếp và hàng đợi

|  |  |  |
| --- | --- | --- |
|  | ****Ngăn Xếp (Stack)**** | ****Hàng Đợi (Queue)**** |
| ****Nguyên Tắc Hoạt Động**** | LIFO (Last In, First Out) | FIFO (First In, First Out) |
| ****Thao Tác Chính**** | **Push** (thêm phần tử vào đỉnh)  **Pop** (lấy phần tử ở đỉnh) | **Enqueue** (thêm phần tử vào cuối)  **Dequeue** (lấy phần tử ở đầu) |
| ****Ứng Dụng**** | Theo dõi thứ tự đảo ngược | Theo dõi thứ tự tuyến tính |
| ****Ví Dụ Thực Tế**** | Ngăn xếp của đĩa trong ngăn tủ | Hàng đợi chờ mua vé tại quầy vé |

1. Định nghĩa cây NP ? Các loại cây NP ?VD? Vẽ cây? Duyệt cây

KN: Cây nhị phân là cây rỗng hoặc là cây mà mỗi nút có tối đa hai nút con. Trong đó các nút con của cây được phân biệt thứ tự rõ ràng, một nút con gọi là nút con trái và một nút con gọi là nút con phải. Ta qui ước vẽ nút con trái bên trái nút cha và nút con phải bên phải nút cha, mỗi nút con được nối với nút cha của nó bởi một đoạn thẳng.

Các loại cây NP đặc biệt



1. Định nghĩa cây NPTK? Vẽ cây VD? Duyệt cây?

Định nghĩa và phân loại cây nhị phân  
Cây nhị phân là cây mà mỗi nút có tối đa hai nút con. Trong đó  
các nút con của cây được phân biệt thứ tự rõ ràng, một nút con  
gọi là nút con trái và một nút con gọi là nút con phải

Vẽ cây VD? Duyệt cây?