Câu 1:

Một cây tìm kiếm nhị phân (Binary Search Tree – viết tắt là BST) là một cây mà trong đó tất cả các nút đều có các đặc điểm sau:

Cây con bên trái của một nút có khóa (key) nhỏ hơn hoặc bằng giá trị khóa của nút cha (của cây con này).

Cây con bên phải của một nút có khóa lớn hơn hoặc bằng giá trị khóa của nút cha (của cây con này).

Một số hoạt động cơ bản có thể được thực hiện trên cây tìm kiếm nhị phân:

Hoạt động tìm kiếm: tìm kiếm một phần tử trong một cây.

Hoạt động chèn: chèn một phần tử vào trong một cây.

Hoạt động duyệt tiền thứ tự: duyệt một cây theo cách thức duyệt tiền thứ tự.

Hoạt động duyệt trung thứ tự: duyệt một cây theo cách thứ duyệt trung thứ tự.

Hoạt động duyệt hậu thứ tự: duyệt một cây theo cách thức duyệt hậu thứ tự.

Câu 2:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ‘ | Khái niệm | Cấu trúc | Các truy xuất | Thời gian tìm kiếm |
| Cây nhị phân tìm kiếm | Cây nhị phân tìm kiếm là cây | struct node  node \*left  node \*right | Truy xuất theo LNR, LRN, NLR. | Nhanh |
| Danh sách đặc | Danh sách đặclà một danh sách mà các phần tử trong danh sách có cùng kiểu dữ liệu,và được cấp phát liên tục trong bộ nhớ. | #define MAX 100  int a[MAX]  int n | Có thể truy xuất ngẫu nhiên các phần tử trong mảng. | Chậm |
| Danh sách liên kết đơn | Danh sách liên kết là danh sách mà các phần tử được cấp phát rời rạc nhau, cố định trong bộ nhớ. Mỗi thành phần trong danh sách gồm có 2 thành phần: vùng thông tin và vùng liên kết. | struct node  {  int info  node \*link  node \*first, \*last  } | Truy xuất tuần tự theo chiều first. | Chậm |
| Danh sách hạn chế | Danh sách hạn chế là danh sách có thứ tự, khi hoạt động truy xuất, phải luôn tuân theo quy tắc đã đề ra, không thể truy xuất phần tử ngẫu nhiên trong danh sách. | Stack và Queue | Truy xuất theo quy tắt FILO và FIFO. | Chậm |