**TRƯỜNG ĐẠI HỌC MỞ TP HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙡❄🙣**



**CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**

### Bài Tập Chương 4 – Câu Hỏi

**GVHD: Thầy Lê Ngọc Hiếu**

|  |  |
| --- | --- |
| **Họ và tên** | **MSSV** |
| **Bùi Đăng Khoa** | **1654050050** |



NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Tp. Hồ Chí Minh, ngày 05 tháng 8 Năm 2019

**Ths. Lê Ngọc Hiếu**

**CÂU HỎI**

**Câu 1: Hãy trình bày các vấn đề sau: Định nghĩa và đặc điểm của Cây nhị phân tìm kiếm; Các thao tác thực hiện tốt trong kiểu này; Hạn chế của kiểu cấu trúc dữ liệu này?**

- Cây nhị phân tìm kiếm là cây nhị phân mà giá trị (khóa) của phần tử bên trái của một node có giá trị nhỏ hơn giá trị của (khóa) của node, giá trị (khóa) của các phần tử bên phải của một node thì lớn hơn giá trị (khóa) của node đó.

- Các thao tác thực hiện tốt trong kiểu cây nhị phân tìm kiếm là:

+ Tìm một node trên cây nhị phân tìm kiếm

+ Thêm một node mới vào cây

+ Duyệt cây nhị phân tìm kiếm

+ Xóa một node trên cây

- Hạn chế của kiểu dữ liệu này là không thể truy xuất một phần tử (node) trực tiếp, mà phải duyệt bắt đầu từ node gốc

**Câu 2: So sánh Cây nhị phân tìm kiếm với các cấu trúc dữ liệu cơ bản như Danh sách đặc, Danh sách liên kết, Danh sách hạn chế.**

* **Giống nhau:**

- Đều thực hiện được các thao tác cơ bản như thêm, chèn, xóa, tìm kiếm, các phép duyệt tương tự các kiểu dữ liệu khác.

- Tương tự như kiểu cấu trúc dữ liệu Danh sách liên kết, Cây nhị phân tìm kiếm cũng có khả năng quản lý một tập các phần tử có số lượng lớn, được cấp phát rời rạc trong bộ nhớ.

- Cây nhị phân tìm kiếm còn có điểm giống với cấu trúc dữ liệu Danh sách liên kết là đều tốn bộ nhớ để lưu trữ các biến con trỏ

* **Khác nhau:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **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 |