



## Cấu trúc dữ liệu và giải thuật

---

### Cuối kì Harmony BTL1 + BTL2

---

nhóm thảo luận CSE  
<https://www.facebook.com/groups/211867931379013>

Tp. Hồ Chí Minh, Tháng 10/2023



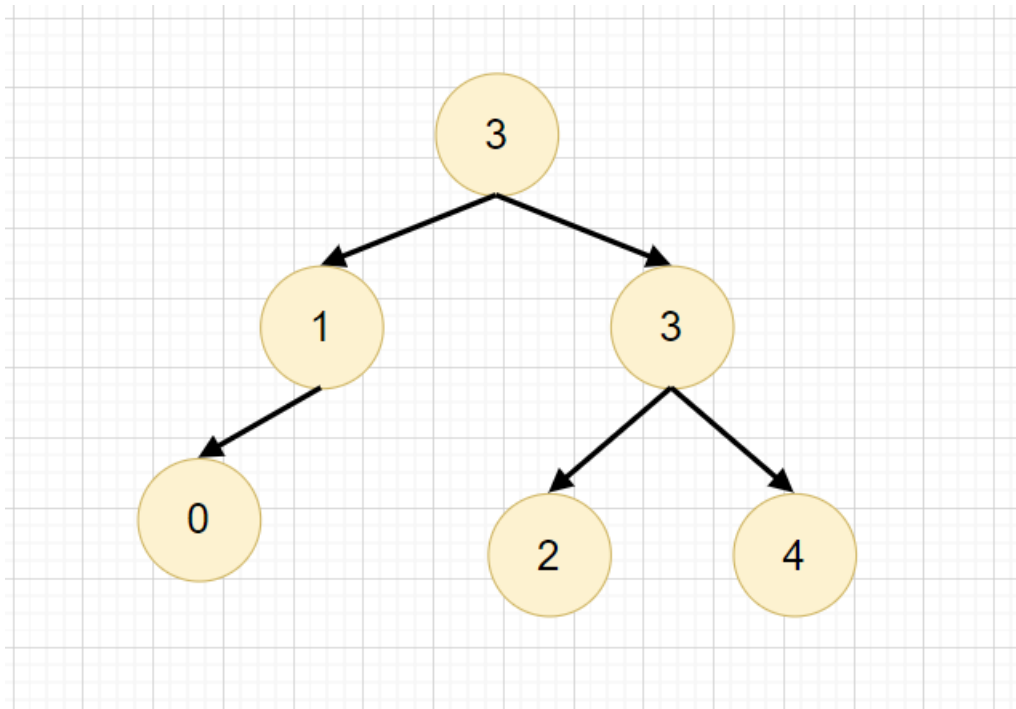
## Mục lục

1	BTL 2	3
---	-------	---



## 1 BTL 2

- [harmony BTL2] Cho cây như hình vẽ tính toán xem có bao nhiêu hoán vị của mảng vừa tạo mà nếu thêm lần lượt các phần tử đó theo thứ tự chỉ mục từ nhỏ đến lớn, có thể sinh ra cùng một cây BST như cây BST gốc.



**Bước 1:** Xác định cây mảng đầu tiên duyệt **BFS** thu được mảng Với số màu đỏ là các phần tử thuộc cây bên trái màu đỏ, ngược lại màu xanh.

- 3 1 3 0 2 4 (1)

**Bước 2:** Tính hoán vị : vì khi duyệt lần lượt từ đầu phần tử đến cuối thì cây con bên trái và bên phải phải đảm bảo thứ tự của từng node nhưng thứ tự so với cây còn lại thì có thể thay đổi được. Vì có 5 vị trí (số node của trái + phải) ta chọn 2 vị trí cho node bên trái vẫn giữ thứ tự là 1 x 0 x x ta có tổ hợp là

$${}_{nodeLeft+nodeRight}C_{nodeLeft} = {}_{nodeLeft+nodeRight}C_{nodeRight}$$

ví dụ trên là  ${}^5C_2 = 10$

- 3 1 0 3 2 4 (2)
- 3 1 3 0 2 4 (3)
- 3 1 3 2 0 4 (4)
- 3 1 3 2 4 0 (5)
- 3 3 1 0 2 4 (6)
- 3 3 1 2 0 4 (7)
- 3 3 1 2 4 0 (8)
- 3 3 2 1 0 4 (9)
- 3 3 2 1 4 0 (10)

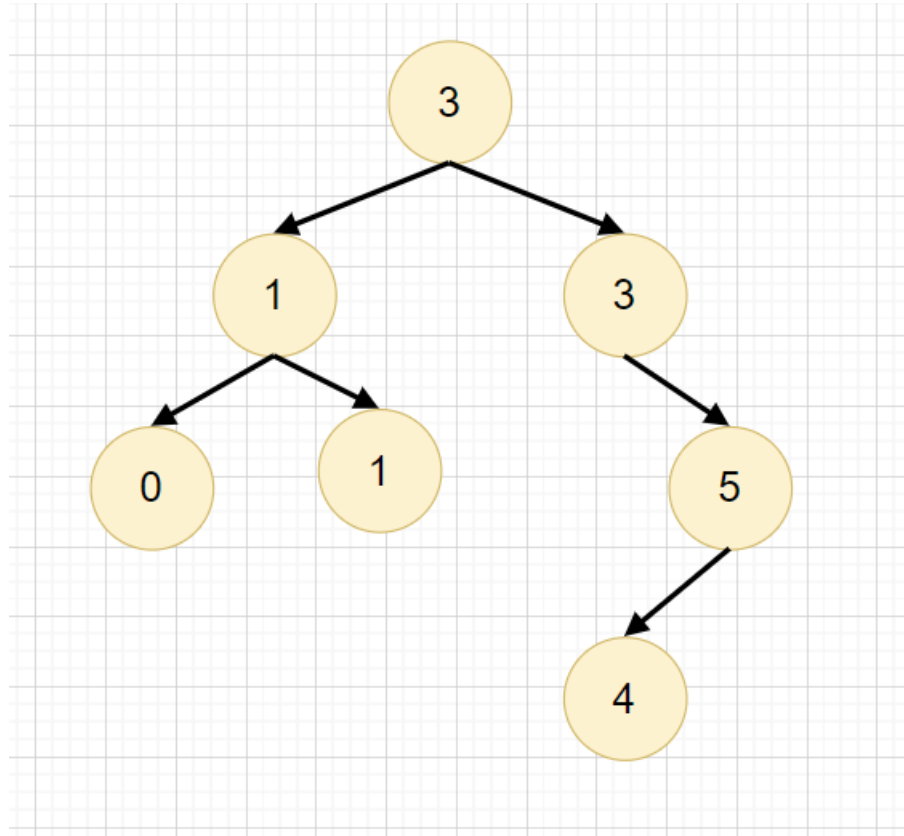
**Bước 3:** Xét đến cây con bên trái (đệ quy) là vì chỉ có 2 nút là 1 và 0 nên chỉ có 1 trường hợp là 1 trước 0 không kết quả DFS này 1

**Bước 4:** Xét đến cây con bên phải (đệ quy) vì có 3 nút nên sẽ có 2 TH là 3 2 4 và 3 4 2 kết quả DFS này là 2



**Bước 5:** kết quả tính được là  ${}^5C_2 * DFS(left) * DFS(right) = {}^5C_2 * 1 * {}^2C_1 * DFS(right \rightarrow left) * DFS(right \rightarrow right) = {}^5C_2 * 1 * {}^2C_1 * 1 * 1 = 20$ , ta có nếu có 2 nút = 1 nút = 1

2. [harmony BTL2] Cho cây như hình vẽ tính toán xem có bao nhiêu hoán vị của mảng vừa tạo mà nếu thêm lần lượt các phần tử đó theo thứ tự chỉ mục từ nhỏ đến lớn, có thể sinh ra cùng một cây BST như cây BST gốc. liệt kê toàn bộ hoán vị có thể xảy ra



3. [harmony BTL2] Cho min heap (xét size là số lượng khách, nếu 2 khách bằng nhau thì khu nào có khách vừa vô sớm nhất thì lớn nhất) như hình vẽ và mảng LRU nói về sự thay đổi khách của các bàn so gần đây nhất  $6- > 7- > 4- > 5- > 2- > 3- > 1$ , hãy liệt kê sách heap các TH sau

ID = 1 size = 3	ID = 2 size = 4	ID = 3 size = 4	ID = 4 size = 5	ID = 6 size = 5	ID = 7 size = 5
--------------------	--------------------	--------------------	--------------------	--------------------	--------------------

**Solution:**

1 Khách vào khu số 8 :

1 khách vào khu số 3 :

2 khách vào khu số 1 :

đuổi toàn bộ khách của khu ít khách nhất và xóa nó :



4. [harmony BTL2] mã hóa huff chuỗi và xây dựng cây AVL của `name = "axcdab"` giống BTL. liệt kê từng bước

**Solution:**

danh sách freq :

cộng dồn và sort theo chiều tăng dần :

nameCaesar :

Từng bước xây cây huffman :

nhị phân nameCaesar :

chuyển sang thập phân :

5. [harmony BTL2] mã hóa huff chuỗi và xây dựng cây AVL của `name = "aabbccxyz"` giống BTL. liệt kê từng bước

**Solution:**

danh sách freq :

cộng dồn và sort theo chiều tăng dần :

nameCaesar :

Từng bước xây cây huffman :

nhị phân nameCaesar :

chuyển sang thập phân :



nhóm thảo luận CSE

<https://www.facebook.com/groups/211867931379013>

CHÚC CÁC EM THI TỐT

