

CÁU TRÚC DỮ LIỆU & GIẢI THUẬT
HOMEWORK 5.4 – Cây AVL, Cây Đỏ-Đen, Cây AA

---oOo---

1. Cây AVL

Cho dãy key: 70, 12, 15, 10, 30, 40, 50, 90, 20, 60, 80, 100

- Hãy tạo cây AVL bằng cách lần lượt thêm các key trên vào cây. Vẽ cây ở những bước có xảy ra thao tác cân bằng. Ghi chú hệ số cân bằng ở mỗi node.
- Lần lượt xóa key=50 và key=60. Vẽ lại cây sau khi xóa các key. Ghi chú hệ số cân bằng ở mỗi node.

2. Cây Red-Black

Cho dãy key: 70, 12, 15, 10, 30, 40, 50, 90, 20, 60, 80, 100

- Hãy tạo cây Red-Black bằng cách lần lượt thêm các key trên vào cây. Vẽ cây ở những bước có xảy ra thao tác cân bằng. Ghi chú màu của mỗi node.
- Lần lượt xóa key=50 và key=60. Vẽ lại cây sau khi xóa các key. Ghi chú màu của mỗi node.

3. Cây AA

Cho dãy key: 70, 12, 15, 10, 30, 40, 50, 90, 20, 60, 80, 100

- Hãy tạo cây AA bằng cách lần lượt thêm các key trên vào cây. Vẽ cây ở những bước có xảy ra thao tác cân bằng. Ghi chú mức (level) của mỗi node.
- Lần lượt xóa key=50 và key=60. Vẽ lại cây sau khi xóa các key. Ghi chú mức (level) của mỗi node.

4. Cây AVL

Cho một cây nhị phân tìm kiếm với gốc là Root (NODE *Root). Hãy viết hàm isAVL để kiểm tra xem cây có phải là cây cân bằng AVL hay không. Prototype của hàm là:

```
// kết quả trả về:  
// + true: cây AVL  
// + false: không phải cây AVL  
bool isAVL(NODE *p);
```

5. Cây Red-Black

Tính chất nào sau đây không phải là của cây Red-Black:

- Mọi đường dẫn từ gốc đến node ngoài phải có cùng số lượng node đen
- $h \leq 2 \cdot h_b$ với h là chiều cao của cây và h_b là chiều cao đen
- $h \leq 1.44 \cdot \log_2 N$ với N là số node
- thời gian tìm kiếm $O(\log_2 N)$

--- Hết ---