**Bài tập cây AVL**

1. Định nghĩa node của cây AVL với kiểu dữ liệu là int và 2 node pLeft và pRight. Lưu ý, ngoài các yếu tố của cây nhị phân, cây AVL còn có thêm một yếu tố cho biết sự cân bằng tại node đó.
2. Viết hàm tạo node, với khuôn mẫu hàm như sau:

Node\* New\_node(int data);

1. Viết hàm xoay trái node p. (Hàm này có thể có giá trị trả về hoặc không, tùy vào cách cài đặt của sinh viên)
2. Viết hàm xoay phải node p. (Hàm này có thể có giá trị trả về hoặc không, tùy vào cách cài đặt của sinh viên)
3. Viết hàm kiểm tra xem, giá trị k có nằm trong cây AVL không? Hàm trả về true/false tương ứng.
4. Viết hàm thêm node có giá trị k vào cây AVL.

\* Gợi ý cách làm (sinh viên có thể tiếp cận theo nhiều hướng khác):

- Thêm node vào cây nhị phân tìm kiếm.

- Kiểm tra xem, cây còn cân bằng hay không.

- Nếu cây không còn cân bằng, thực hiện các phép xoay để cây trở nên cân bằng.

1. Viết hàm xóa node có giá trị k từ cây AVL

**Quy định nộp bài:**

* Sinh viên tổ chức thư mục theo định dạng:
  + MSSV
    - Code: các file \*.cpp giải các bài tập: 1.cpp, 2.cpp, …, 6.cpp
    - Report
      * File report.pdf trả lời các câu hỏi trong file bài tập. (Không cần làm câu 8 trong phần yêu cầu, và câu số 3 trong phần áp dụng nâng cao).
      * source.cpp: chứa mã nguồn tương ứng của file report trên
* Tất cả được đặt trong thư mục MSSV, nén lại thành MSSV.zip hoặc MSSV.rar
* Các bài làm giống nhau sẽ bị 0 điểm môn học.