

Ta thấy quá trình cắt đoạn gỗ ban đầu thành N thanh và quá trình ghép N thanh thành 1 thanh gỗ duy nhất với chi phí ghép hai thanh chính là tổng hai thanh đó đều cho ra kết quả như nhau.

Một điều nữa:

- Với $N=2$ thì chi phí luôn là tổng hai thanh con.
- Với $N=3$ thì chi phí bằng chi phí ghép hai thanh với thanh còn lại. Mà chi phí ghép với thanh còn lại luôn cố định (bằng độ dài thanh ban đầu), còn chi phí ghép hai thanh trước đó có thể thay đổi. Mà để kết quả thu được nhỏ nhất thì tổng hai thanh này nhỏ nhất. Suy ra ta chọn hai thanh ngắn nhất ghép với nhau rồi sau đó ghép với thanh thứ 3 thì sẽ ra kết quả tối ưu.
- Từ đó ta suy ra trường hợp $N=k$: chọn hai thanh ngắn nhất trong các thanh này ghép lại được một thanh mới, số thanh giảm đi 1, còn lại $k-1$ thanh. Tiếp tục ghép theo cách như vậy cho đến khi được thanh ban đầu thì ta có chi phí nhỏ nhất.

Vậy làm sao để chọn ra hai thanh ngắn nhất trong các thanh hiện có trong thời gian cho phép?

Ta sử dụng heap để lưu trữ độ dài các thanh gỗ và ưu tiên các thanh ngắn gần gốc hơn (gốc chứa thanh có độ dài nhỏ nhất). Mỗi khi đọc độ dài một thanh gỗ ta cập nhật vào heap. Sau đó ta thực hiện thao tác ghép lại: lấy ra hai thanh ngắn nhất, ghép hai thanh này lại (kết quả tăng lên chính là tổng độ dài hai thanh) và đẩy thanh mới ghép vào heap.

Độ phức tạp thuật toán: $n \log n$

Tham khảo lời giải các bài khác hoặc thảo luận ngay tại đây: <https://icnhoukdsiih.blogspot.com/>