Nhóm 8:

Nếu c quá lớn, thì có ảnh hưởng lớn đến thuật toán, vì vậy chặn trên là bao nhiêu?

* Nếu c <= 10 thì có thể bỏ được còn lớn hơn thì không.

Cách tính độ phức tạp có phụ thuộc vào ngôn ngữ hay không?

* Không dựa vào nnlt, dựa vào số lần phép tính

Khác nhau giữa độ phức tạp xấu nhất và trung bình

* Xấu nhất là trường hợp tệ nhất phải đối mặt, và đây chỉ là tương đối, còn phụ thuộc vào số lượng input

Nhóm 9:

Có thể dùng code để kiểm tra đô phức tạp của code?

* Có thể

Nhóm 4:

O(logn) thì log là cơ số mấy? 2 hay 10?

* Các log trong tính độ phức tạp là log2 .

Tính độ phức tạp tgian là dựa vào số phép tính, còn không giạn thì dựa vào điều nào?

* Dựa vào số biến được đặt ra trong bài toán, tuy nhiên độ phức tạp thời gian không được chú ý nhiều.

Độ phức tạp logn với n mà logn < n, nếu input lớn thì nó có sai không?

Nhóm 2:

Một vài ví dụ có độ phức tạp là O(n!)

Nhóm 1:

Ngoài 3 quy tắc ở trên thì còn những quy tắc nào không?

* Các quy tắc đó được dựa trên những tính chất được rút ra từ toán học.

Nhận xét:

* Hình thức slide: chưa đạt, vì hình còn nhỏ, bố cục chưa cân đối.
* Dẫn dắt: người thuyết trình giàu năng lượng, nhưng chưa trả lời được nhiều câu hỏi.

Sửa slide:

* Làm rõ giữa các trường hợp trong độ phức tạp, các ký pháp, big Oh, big theta, …
* Để tính độ phức tạp bằng lý thuyết thì đếm số phép toán cơ sở: tham số và kích thước dữ liệu.(tham số quyết định đpt, xác định phép toán cơ sở)
* Tính số phép toán cơ sở T, tính tỉ suất tăng giữa 2 thuật toán lấy lim giữa 2 T của hai thuật toán, chặn trên và chặn dưới.