

ANHSANG

Nguồn bài: Light - Timus 1464

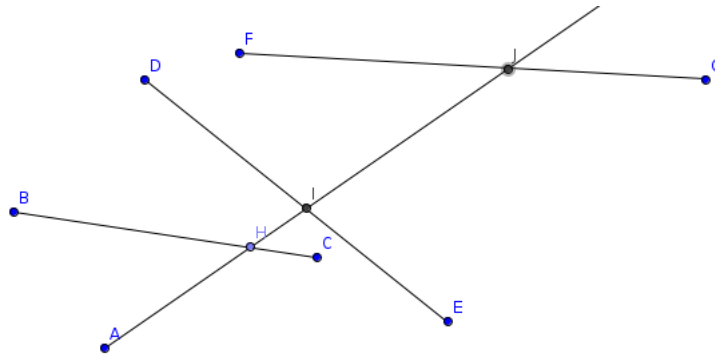
Trường hợp đa giác lồi (10 điểm)

Một đa giác được coi là lồi khi và chỉ khi lấy hai điểm nằm hoàn toàn trong đa giác, đoạn thẳng nối hai điểm đó cũng nằm hoàn toàn trong đa giác. Từ định nghĩa này, ta thấy rằng việc điểm sáng được đặt ở đâu không quan trọng, đáp án luôn là diện tích đa giác được cho.

Trường hợp đa giác lõm (40 điểm)

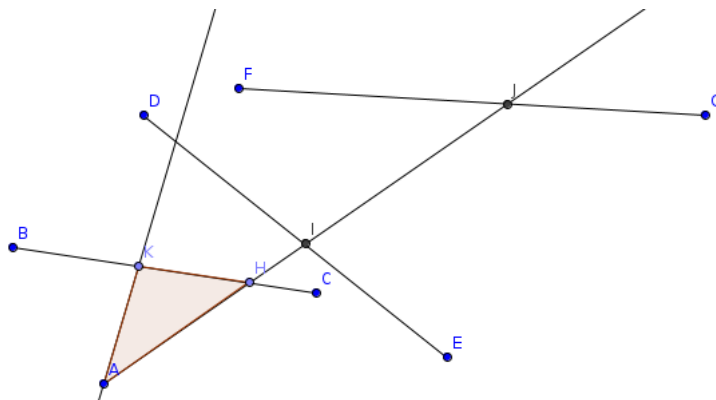
Từ điểm sáng được cho, ta vẽ một tia sáng rồi quét tia sáng đó quanh điểm sáng ban đầu.

Trong quá trình quét, ta duy trì một tập S chứa các cạnh của đa giác cắt tia sáng đang được xét, các đoạn trong tập được sắp xếp theo khoảng cách giữa đoạn thẳng và điểm sáng tính theo tia sáng được xét. Ví dụ, ở hình dưới đây (điểm sáng A , tia sáng AH , các đoạn BC , DE , FG là các cạnh cắt tia sáng), các đoạn trong tập được sắp xếp theo thứ tự BC , DE , FG vì $AH < AI < AJ$



Do đa giác không tự cắt, thứ tự giữa hai đoạn bất kì trong tập S sẽ không bao giờ thay đổi. Nói cách khác, nếu đoạn a nằm trước đoạn b trong tập S ở một thời điểm nào đó thì đoạn b sẽ không bao giờ có thể nhảy lên trước đoạn a trong quá trình quét. Từ tính chất này, ta chỉ cần dùng cấu trúc dữ liệu **set** với một hàm so sánh tự viết là có thể duy trì được tập S .

Ta thấy trong các khoảng thời gian trong quá trình quét mà đoạn đầu tiên của tập S không thay đổi thì ta có thể dễ dàng tính được phần diện tích được chiếu sáng trong khoảng đó. Ví dụ như khi ta quét từ tia AH đến tia AK ở hình dưới đây.



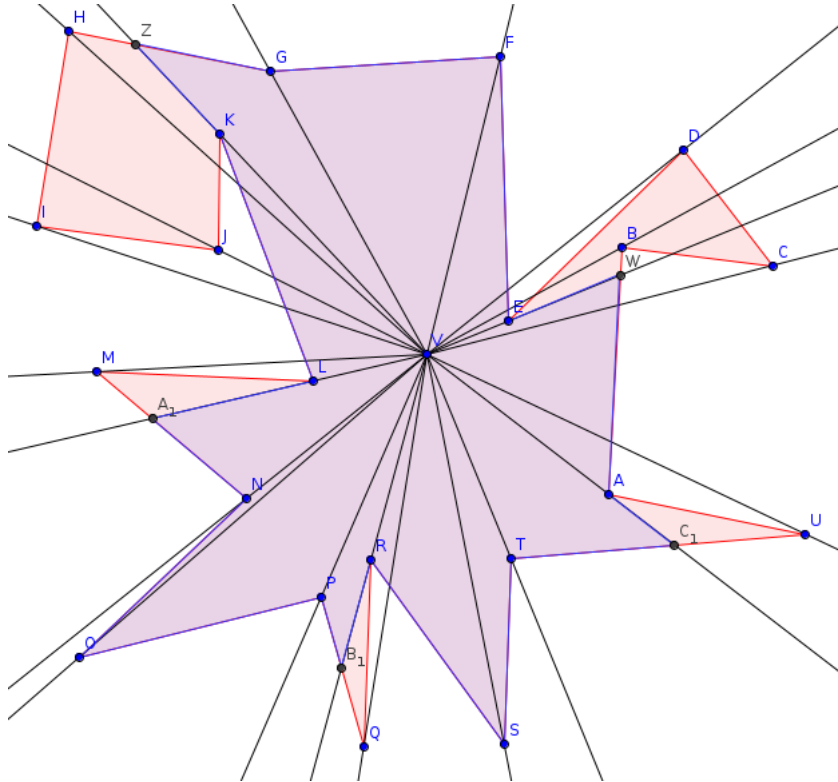
thì phần diện tích chiếu sáng được cộng vào đáp án là diện tích tam giác AKH . Ta có thể dễ dàng tính được diện tích đa giác này.

Vậy thì phần tử đoạn đầu tiên trong tập S thay đổi khi nào? Đó là khi mà đường quét của ta chạm vào một đỉnh nào đó của đa giác. Khi đó, sẽ có một số đoạn phải bị loại bỏ khỏi tập S , và một số đoạn có thể được thêm vào tập S và rất có thể ta có thể có một đoạn đầu tiên mới.

Tóm lại, ta sẽ có ý tưởng thuật toán sơ lược sau:

- Sắp xếp các đỉnh của đa giác theo thứ tự ngược chiều kim đồng hồ
- Xét mỗi đỉnh của đa giác theo thứ tự đã sắp xếp
 - Cập nhật tập S như đã mô tả ở trên
 - Tính phần diện tích chiếu sáng thêm và cộng vào kết quả.

Mình xin gửi một hình mô phỏng lại thuật toán trên. Khoảng giữa hai tia sáng liên tiếp trong hình là một khoảng mà đoạn đầu tiên trong tập S không thay đổi. Phần diện tích màu xanh tím là phần diện tích được chiếu sáng. Hy vọng hình này có thể giúp bạn trong việc gỡ lỗi chương trình.



Kĩ thuật duy trì tập S có tính chất như trên được gọi là **Z-buffering**. Kĩ thuật này thường xuyên được sử dụng trong các trò chơi bắn súng góc nhìn thứ nhất như Half-Life để xác định những vật thể người chơi có thể nhìn thấy mà cần được hiển thị trên màn hình.