**ROBOT**

Nhận xét đầu tiên là ta sẽ không bao giờ đi lên hay xuống trên 1 cột khi không cần thiết. Do nhảy sang cột khác nếu có thể, hạ cùng lượng độ cao, mất cùng lượng chi phí, lại được chuyển cột khác.

Khi nào là cần thiết. Ta sẽ có 3 cách nhảy:

Từ cột tại chiều cai , giả sử nhảy bằng 1 cạnh sang cột tới chiều cao

* Nếu thì ta sẽ đi lên ở cột một đoạn rồi mới nhảy sang cột ở chiều cao 0, chi phí là - 0 + lượng đi lên.
* Nếu ta sẽ đi xuống ở cột một đoạn rồi mới nhảy sang cột ở chiều cao , chi phí là .
* Nếu , nhảy trực tiếp sang, chi phí là .

Từ đây có 1 số nhận xét:

* Nếu dùng chiến thuật trên, khi xuất phát từ tại , đến được tại thì chi phí là .
* Nếu đến được tại chiều cao thì chi phí còn lớn hơn nữa.
* Khi đã ở thì đi tiếp vẫn luôn luôn ở chiều cao

Vậy nên đến được tại vị trí càng cao thì chi phí càng nhỏ.

Rút ra thuật toán là Dijkstra, là chi phí ít nhất tới được đỉnh u, tại chiều cao nhất có thể. (do càng cao chi phí càng ít ).

Luôn giữ lại chiều cao lớn nhất tại cùng một cột.

Độ phức tạp: