Ý tưởng là ta sẽ duyệt từng độ cao, và đi tìm chi phí nhỏ nhất để khôi phục lại bức tường đến độ cao đó. Kết quả của bài toán chính là giá trị nhỏ nhất của những chi phí khôi phục đó.

Trước hết, ta cần tìm ra cách để khôi phục bức tường đến độ cao . Ta sẽ sắp xếp mảng độ cao của các cột theo thứ tự không giảm. Sau đó thực hiện các thao tác để giảm đi số viên gạch ở những cột cao hơn , và tăng thêm số viên gạch ở những cột thấp hơn . Lưu ý rằng, ta không được tạo thêm cột hay bỏ qua những cột có độ cao là .

Ta thấy rằng có 2 trường hợp là:

1. : tổng chi phí đặt và gỡ ít hơn hoặc bằng chi phí di chuyển, ta nên gỡ đi những viên gạch ở cột cao hơn và đặt thêm vào những cột thấp hơn.
2. : tổng chi phí đặt và gỡ nhiều hơn phí di chuyển, ta nên di chuyển những viên gạch ở cột cao hơn sang cột thấp hơn.

Ta cần tìm vị trí sao cho: độ cao các cột từ đều , độ cao các cột từ đều . Tiếp đến ta sử dụng mảng cộng dồn để tính xem từ có tổng cộng bao nhiêu viên gạch.

Gọi là tổng số viên gạch cần thêm vào những cột thấp hơn , ta có công thức:

Và là tổng số viên gạch cần bớt đi ở những cột cao hơn , ta có công thức:

Từ đây ta có các trường hợp khi khôi phục bức tường đến độ cao :

1. Nếu thì chi phí khôi phục tối ưu sẽ là: .
2. Nếu thì sau khi thực hiện các thao tác đặt, gỡ và chuyển thì tổng số viên gạch cần bớt đi ở những cột cao hơn chỉ còn , nên chi phí khôi phục là: .
3. Nếu thì cũng tương tự, còn viên gạch cần thêm ở những cột thấp hơn , nên chi phí khôi phục là: .

Ta đã thực hiện những công việc sau: sắp xếp là , duyệt từng độ cao là với là hằng số chiều cao lớn nhất của các cột ban đầu, tìm kiếm vị trí là , và sau đó tính chi phí khôi phục trong . Vậy độ phức tạp của thuật toán này là .

Với thì thuật toán trên không thể nào được full điểm. Ta mất chi phí nhiều nhất ở việc duyệt qua từng độ cao , vậy nên ta phải tối ưu bằng cách chỉ duyệt những độ cao cần thiết.

Khi duyệt từng độ cao từ và in ra chi phí khôi phục bức tường đến độ cao tương ứng thì ta thấy rằng, chi phí khôi phục có xu hướng giảm (*không nghiêm ngặt*) đến một giá trị nào đó rồi lại tăng lên (*không nghiêm ngặt*). Nghĩa là đường biểu diễn chi phí của từng độ cao là một đường cong đi xuống rồi đi lên. Vậy kết quả của bài toán chính là điểm cực tiểu của đường cong đó.

Ta tìm điểm cực tiểu đó bằng tìm kiếm nhị phân. Ở đây, ta chọn cận dưới và cận trên để đảm bảo là luôn có kết quả.

Ta có là độ cao đang xét.

* Nếu chi phí khôi phục độ cao đều nhỏ hơn các chi phí khôi phục độ cao thấp hơn và cao hơn , thì chính là độ cao mà cần chi phí khôi phục nhỏ nhất.
* Nếu chi phí khôi phục độ cao cao hơn những độ cao thấp hơn, và nhỏ hơn những độ cao thấp hơn, thì có nghĩa là đường cong tại đó đang đi lên. Vậy ta sẽ tìm kiếm tiếp trong đoạn .
* Ngược lại thì đường cong đang đi lên, ta sẽ tìm tiếp trong đoạn

Tới đây ta đã tối ưu bằng cách chỉ xét những độ cao cần thiết. Độ phức tạp của thuật toán chỉ còn .