GIỚI HẠN BÀI TẬP PHẦN STACK + QUEUE

Bỏ qua các bài tập Trung tố, Hậu tố, Tiền tố

Tập trung vào bài toán xử lý cặp dấu ngoặc (hoặc tương tự)

ỨNG DỤNG STACK ĐỂ **KIỂM SOÁT CẶP DẤU NGOẶC**

* Nếu gặp dấu mở ngoặc ( ta push, ngược lại gặp đóng ) thì ta pop:
  + Giúp mình kiểm soát được cặp dấu ngoặc đúng
* Khi nào biểu thức đúng:
  + Khi không còn dấu ngoặc nào thừa
* Nếu biểu thức **không đúng**:
  + Sẽ có biểu thức đúng dài nhất: tính toán được biểu thức đúng dài nhất khi push và pop vị trí của dấu ngoặc
  + Sẽ tính được số dấu ngoặc phải đổi chiều
    - Nếu gặp dấu đóng mà không thấy dấu mở: đổi chiều và push
    - Nếu không còn dấu đóng nào mà bên trong stack vẫn còn N dấu mở => đảo chiều N/2 dấu
  + …
* Phá ngoặc để tìm biểu thức tương đương (\*)

HÀNG ĐỢI

ỨNG DỤNG HÀNG ĐỢI: DUYỆT BFS

Giá trị lớn nhất của kỹ thuật BFS: **số bước chuyển ít nhất** từ trạng thái xuất phát đến trạng thái đích.

Vấn đề:

* Các trạng thái có bị trùng lặp hay không?
  + Nếu không bao giờ bị trùng lặp: BFS dễ: không cần đánh dấu
  + Nếu có thể bị trùng lặp: cần phải đánh dấu các trạng thái đã đi qua
    - Cách 1 (trường hợp đơn giản): đánh dấu bằng mảng
    - Cách 2 (trường hợp khó): đánh dấu bằng **set** hoặc **map**

PHÂN LỚP BÀI TOÁN

Dễ nhất: BFS không cần đánh dấu

* BFS nhị phân (09, BDN, lộc phát, …)

Trung bình: BFS đánh dấu bằng mảng

* Biến đổi 0-1
* Biến đổi S-T
* Di chuyển trong ma trận.

Khó: BFS đánh dấu bằng set và map (bỏ qua)