

Hàng đợi - Queue

Hàng đợi (tiếng Anh: queue) là một cấu trúc dữ liệu dùng để chứa các đối tượng làm việc theo cơ chế FIFO (viết tắt từ tiếng Anh: First In First Out), nghĩa là việc thêm vào hoặc lấy một đối tượng ra khỏi hàng đợi, được thực hiện theo cơ chế "vào trước ra trước".

Cấu trúc dữ liệu hàng đợi có thể định nghĩa như sau: Hàng đợi là một cấu trúc dữ liệu trừu tượng (ADT) tuyến tính. Tương tự như ngăn xếp, hàng đợi hỗ trợ các thao tác:

EnQueue(o): thêm đối tượng o vào cuối hàng đợi.

DeQueue(): lấy đối tượng ở đầu queue ra khỏi hàng đợi và trả về giá trị của nó. Nếu hàng đợi rỗng thì lỗi sẽ xảy ra.

IsEmpty(): kiểm tra xem hàng đợi có rỗng không.

Front(): trả về giá trị của phần tử nằm ở đầu hàng đợi mà không hủy nó. Nếu hàng đợi rỗng thì lỗi sẽ xảy ra.

Các thao tác thêm, trích và hủy một phần tử phải được thực hiện ở hai phía khác nhau của hàng đợi, do đó hoạt động của hàng đợi được thực hiện theo nguyên tắc FIFO. Cũng như ngăn xếp, cấu trúc mảng một chiều hoặc cấu trúc danh sách liên kết có thể dùng để biểu diễn cấu trúc hàng đợi.



(c) sunlight

Ứng dụng:

Cho đơn đồ thị vô hướng $G = (E, V)$, có n đỉnh m cạnh. Một đường đi từ đỉnh s đến đỉnh t là 1 dãy các đỉnh không trùng nhau $s = x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_k = t$. Độ dài đường đi là số cạnh trên đường đi

Yêu cầu: Tìm đường đi ngắn nhất từ đỉnh s đến đỉnh t .

Hint:

Sử dụng thuật toán tìm kiếm theo chiều rộng - BFS sử dụng cấu trúc Queue. Thuật toán sẽ có cấp độ $O(M+N)$ hoặc $O(N^2)$ tùy cách cài đặt. Ngoài ra cũng có thể sử dụng 1 trong số các thuật toán tìm đường đi ngắn nhất khác.

Input

Dòng 1: Gồm 4 số n, m, s, t ($1 \leq n \leq 100$)

M dòng tiếp theo: Mỗi dòng gồm 2 số i, j mô tả 1 cạnh của đồ thị.

Output

Dòng 1: ghi độ dài đường đi ngắn nhất.

Dòng 2: Ghi các đỉnh theo thứ tự trên đường đi từ s đến t .

Example

Input:

```
5 5 1 5
1 3
3 5
1 4
4 2
2 5
```

Output:

```
2
1 3 5
```