

Bài tập đồ thị

* Lưu ý: đồ thị trong bài tập trên là đồ thị có hướng (directed graph)

0. Định nghĩa một struct hoặc một class dùng để lưu trữ đồ thị có V đỉnh.
(Khuyến khích các bạn viết theo hướng đối tượng, vì có thể tạo một đối tượng Graph và thao tác trên đối tượng đó).
1. Viết chương trình cho phép người dùng nhập vào các cạnh của đồ thị và biểu diễn đồ thị dưới dạng một ma trận kề (adjacency matrix).

Ví dụ: khi người dùng nhập vào:

```
graph.add_edge(0, 1)
```

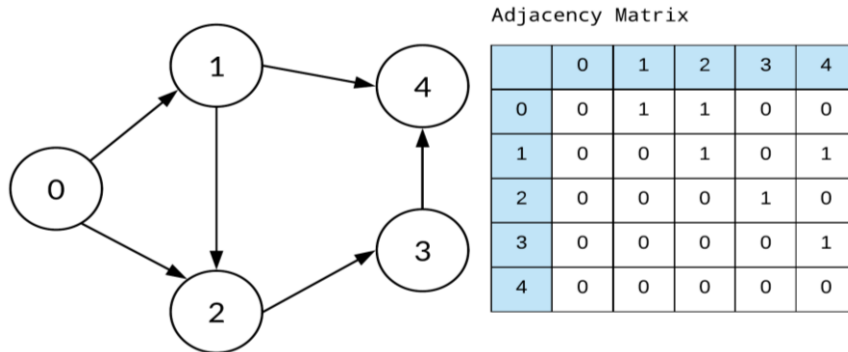
```
graph.add_edge(0, 2)
```

```
graph.add_edge(1, 2)
```

```
graph.add_edge(1, 4)
```

```
graph.add_edge(2, 3)
```

```
graph.add_edge(3, 4)
```



2. Deep first search (duyệt theo chiều sâu)
 - a. Cài đặt hàm duyệt qua tất cả các node của đồ thị, dùng kỹ thuật deep first search.
 - b. Nhận xét độ phức tạp về không gian và thời gian của deep first search.
3. Breadth first search (duyệt theo chiều rộng)
 - a. Cài đặt hàm duyệt qua tất cả các node của đồ thị, dùng kỹ thuật breadth first search.
 - b. Nhận xét độ phức tạp về không gian và thời gian của kỹ thuật breadth first search.

4. Viết hàm kiểm tra xem, giữa hai đỉnh có tồn tại đường đi hay không? Nếu tồn tại, hãy in đường đi ra màn hình. (Sinh viên có thể dùng DFS hoặc BFS tùy ý)

Quy định nộp bài:

- Các bạn nộp file 1.cpp, 2.cpp, 3.cpp, 4.cpp tương ứng. Đối với câu 2b và 3b sinh viên chỉ cần ghi chú trong source code.
- Nén lại thành MSSV.zip hoặc MSSV.rar