

[Dashboard](#) / [My courses](#) / [Graph Theory-HK3-0405](#) / [Tuần 8 - Thứ tự topo & Ứng dụng](#) / [001b. Thứ tự topo \(chiều rộng hoặc chiều sâu\)](#)

Started on	Tuesday, 1 July 2025, 4:21 PM
State	Finished
Completed on	Tuesday, 1 July 2025, 4:29 PM
Time taken	8 mins 39 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)

Question 1

Correct

Mark 1.00 out of 1.00

Viết chương trình đọc vào một *đồ thị có hướng không chu trình* G , in các đỉnh của G ra màn hình theo thứ tự topo. Nếu có nhiều thứ tự topo, in một thứ tự bất kỳ.

Đầu vào (Input)

Dữ liệu đầu vào được nhập từ bàn phím với định dạng:

- Dòng đầu tiên chứa 2 số nguyên n và m , tương ứng là số đỉnh và số cung.
- m dòng tiếp theo mỗi dòng chứa 2 số nguyên u, v mô tả cung (u, v) .

Đầu ra (Output)

- In các đỉnh ra màn hình theo thứ tự topo. In các đỉnh trên một dòng, cách nhau 1 khoảng trắng.

Gợi ý

Ngoài giải thuật sắp xếp topo dựa trên phương pháp duyệt theo chiều rộng, ta cũng có thể sắp xếp topo dựa trên duyệt theo chiều sâu.

```
void dfs(G, u, L) {
    mark[u] = 1;
    for (v là kề của u)
        if (mark[v] == 0)
            dfs(G, v, L);
    push_back(L, u);
}
```

Khởi tạo $mark[u] = 0$ với mọi $u = 1, n$

```
for (u = 1; u <= n; u++)
```

```
    if (mark[u] == 0)
```

```
        dfs(G, u, L);
```

In ngược các phần tử của L

Answer: (penalty regime: 10, 20, ... %)

```
1 #include <stdio.h>
2 #define MAX_N 100
3
4 //_____LIST_____
5 typedef struct
6 {
7     int data[MAX_N]; // mảng bao gồm các phần tử của danh sách
8     int size;        // độ dài của danh sách
9 } List;
10
11 void make_null_list(List *pL)
12 {
13     pL->size = 0;
14     // (*L).size=0;
15 }
16
17 // Thêm một phần tử mới vào cuối ds
18 void push_back(List *pL, int x)
19 {
20
21     // pL->data[pL->size] = x;
22     // pL->size++;
```

Test	Got
✓ Sinh test và chấm tự động	<div>✓</div> <p>Test 1:</p> <p>10 15</p> <p>3 8</p> <p>4 1</p> <p>1 8</p> <p>5 1</p> <p>2 7</p> <p>10 7</p> <p>1 7</p> <p>7 8</p> <p>5 3</p> <p>9 8</p> <p>10 3</p> <p>10 2</p> <p>1 2</p> <p>4 3</p> <p>5 4</p> <p>Kết quả chương trình: 5 6 9 10 4 1 3 2 7 8</p> <p>ĐÚNG</p> <p>-----</p> <p>Test 2:</p> <p>9 12</p> <p>9 3</p> <p>4 7</p> <p>3 2</p> <p>6 9</p> <p>7 1</p> <p>9 5</p> <p>5 3</p> <p>8 9</p> <p>7 2</p> <p>1 5</p> <p>8 5</p> <p>8 7</p> <p>Kết quả chương trình: 4 6 8 7 9 1 5 3 2</p> <p>ĐÚNG</p> <p>-----</p> <p>Test 3:</p> <p>10 13</p> <p>2 7</p> <p>7 8</p> <p>5 10</p> <p>3 10</p> <p>7 9</p> <p>4 10</p> <p>10 9</p> <p>1 7</p> <p>4 7</p> <p>9 8</p> <p>3 6</p> <p>4 9</p> <p>3 2</p> <p>Kết quả chương trình: 1 3 4 5 2 6 10 7 9 8</p> <p>ĐÚNG</p> <p>-----</p> <p>Test 4:</p> <p>7 9</p> <p>7 4</p> <p>2 5</p> <p>3 5</p> <p>7 5</p> <p>4 6</p> <p>2 3</p> <p>7 6</p> <p>5 6</p> <p>2 7</p> <p>Kết quả chương trình: 1 2 3 7 4 5 6</p> <p>ĐÚNG</p> <p>-----</p> <p>Test 5:</p> <p>7 10</p> <p>7 4</p> <p>4 3</p> <p>7 2</p> <p>3 2</p> <p>5 2</p>

Test	Got
	1 5 2 6 1 6 4 2 5 4 Kết quả chương trình: 1 7 5 4 3 2 6 ĐÚNG -----

Passed all tests! ✓

Question author's solution (C):

```

1  #include <stdio.h>
2
3  #define MAX_N 100
4  typedef struct {
5      int n, m;
6      int A[MAX_N][MAX_N];
7  } Graph;
8
9
10 void init_graph(Graph *pG, int n) {
11     pG->n = n;
12     pG->m = 0;
13     for (int u = 1; u <= n; u++)
14         for (int v = 1; v <= n; v++)
15             pG->A[u][v] = 0;
16 }
17
18 void add_edge(Graph *pG, int u, int v) {
19     pG->A[u][v] += 1;
20 }
21
22

```

Correct

Marks for this submission: 1.00/1.00.

◀ 001. Thứ tự topo (chiều rộng)

Jump to...

002. Xếp đá ▶