

Bicoloring (BFS) — [Bicoloring problem in Onlinejudge](#)

Problem: Bicoloring (BFS)

Problem: Bicoloring (BFS)

সমাধান সংক্ষেপণ

আমাদের কাজ হলো একটি গ্রাফকে দুটি রঙে রঙ করা –
যাতে কোনো দুটি সংযুক্ত (adjacent) নোডের রঙ কখনও এক না হয়।

যদি এটা করা সম্ভব হয় \rightarrow গ্রাফটি Bicolorable

না হলে \rightarrow Not Bicolorable

এটিই হলো Graph Bipartite Checking Problem

বাস্তব উদাহরণ

ভাবো, তুমি একটা মানচিত্র রঙ করছো –
যে কোনো দুটি পাশাপাশি দেশের রঙ যেন এক না হয়।
এখানে প্রতিটি দেশ = Node, এবং সীমান্ত = Edge।

যদি দুইটা রঙে (ধরা যাক, কালো ও সাদা) এমনভাবে রঙ করা সম্ভব হয় \rightarrow
তাহলে মানচিত্র বা গ্রাফটি Bipartite / Bicolorable

মূল ধারণা: Bipartite Graph

একটা গ্রাফ Bipartite হয় যদি তুমি নোডগুলোকে দুটি সেটে ভাগ করতে পারো (ধরা যাক A এবং B) –
যেখানে:

- প্রতিটি edge সবসময় A \rightarrow B তে যায়
- A-এর কোনো দুই নোড নিজেদের মধ্যে যুক্ত না
- B এর কোনো দুই নোড নিজেদের মধ্যে যুক্ত না

কখন দুই রঙে রঙ করা যাবে না?

যদি গ্রাফে কোনো বিজোড় দৈর্ঘ্যের সাইকেল (Odd-Length Cycle) থাকে,
তাহলে সেটা কখনও দুই রঙে রঙ করা যাবে না।

সমাধান: BFS দিয়ে

আমরা Breadth First Search (BFS) দিয়ে গ্রাফটাকে ট্রাভার্স করবো এবং প্রতিটি নোডে রঙ বসাবো।

ধাপসমূহ:

- সব নোডকে শুরুতে uncolored (-1) হিসেবে ধরো।
- প্রথম নোড (যেমন 0) থেকে BFS শুরু করো \rightarrow color[0] = 0
- প্রতিবার queue থেকে একটা নোড বের করো
 - তার প্রতিবেশীদের বিপরীত রঙে রঙ করো

- যদি কোনো প্রতিবেশীর রঙ আগেই দেওয়া থাকে এবং সেটি একই রঙ হয়, তাহলে
→ Conflict, মানে গ্রাফটি দুই রঙে রঙ করা যাবে না।
4. যদি BFS শেষ হয় এবং কোনো conflict না পাওয়া যায় → BICOLORABLE

স্যাম্পল ইনপুট (ত্রি ভূজ) ওয়াক্স্ট:

ইনপুট:

3

3

0

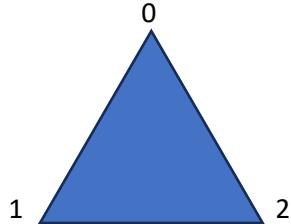
1

1

2

2

0



১. color অ্যারে = [-1, -1, -1]

২. নোড ০ থেকে শুরু করি | color[0] = 0 | queue: [0] | color = [0, -1, -1]

৩. queue থেকে ০ বের করি (u = 0, currentColor = 0, neighborColor = 1) |

৪. 0-এর প্রতিবেশী 1: রঙবিহীন | color[1] = 1 | queue: [1] | color = [0, 1, -1]

৫. 0-এর প্রতিবেশী 2: রঙবিহীন | color[2] = 1 | কিউ: [1, 2] | color = [0, 1, 1]

৬. queue থেকে 1 বের করি (u = 1, currentColor = 1, neighborColor = 0) |

৭. 1-এর প্রতিবেশী 0: color[0] (যা 0) currentColor (যা 1)-এর সমান নয়। ঠিক আছে।

৮. 1-এর প্রতিবেশী 2: color[2] (যা 1) currentColor (যা 1)-এর সমান!

৯. Conflict! দাটু সংলগ্ন নোড (১ এবং ২) একই রঙ (1) পেয়েছে।

১০. অ্যালগরিদম থামবে।

আউটপুট: NOT BICOLORABLE.

Pseudocode

```
function BFS_Bicoloring(Graph, n):

    color[n] = {-1} // initially uncolored
    for each node u in Graph:
        if color[u] == -1:
            color[u] = 0
            queue.push(u)

    while queue not empty:
        x = queue.front()
        queue.pop()

        for each neighbor y of x:
            if color[y] == -1:
                color[y] = 1 - color[x]
                queue.push(y)

            else if color[y] == color[x]:
                return "NOT BICOLORABLE"
    return "BICOLORABLE"
```