

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.
- For each connected component, mark the labels that it contains by traversing it once and save the indices of the associated nodes.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.
- For each connected component, mark the labels that it contains by traversing it once and save the indices of the associated nodes.
- If, while traversing, a node with an already marked label is found, the problem is done: output the index of the node with the previously marked label and the index of the newly found node.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.
- For each connected component, mark the labels that it contains by traversing it once and save the indices of the associated nodes.
- If, while traversing, a node with an already marked label is found, the problem is done: output the index of the node with the previously marked label and the index of the newly found node.
- If all connected components have been checked and no matching labels were found, output 'safe'.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.
- For each connected component, mark the labels that it contains by traversing it once and save the indices of the associated nodes.
- If, while traversing, a node with an already marked label is found, the problem is done: output the index of the node with the previously marked label and the index of the newly found node.
- If all connected components have been checked and no matching labels were found, output 'safe'.
- Time complexity: $\mathcal{O}(n + m)$.

E: Ernest's Endeavour

Problem author: David Ghiberdic, Rares Rauta

- **Problem:** Given an undirected graph with n labeled nodes and m edges, check if any two nodes with the same label are connected.
- Naive solution: For each node, traverse the graph until you reach another node with the same label.
- Time complexity: $\mathcal{O}(n \cdot (n + m))$ - too slow!
- **Key insight:** Two nodes are connected iff they are part of the same connected component.
- Therefore, we only need to check if a connected component of the graph has two nodes with identical labels.
- For each connected component, mark the labels that it contains by traversing it once and save the indices of the associated nodes.
- If, while traversing, a node with an already marked label is found, the problem is done: output the index of the node with the previously marked label and the index of the newly found node.
- If all connected components have been checked and no matching labels were found, output 'safe'.
- Time complexity: $\mathcal{O}(n + m)$.

Statistics: ... submissions, ... accepted, ... unknown