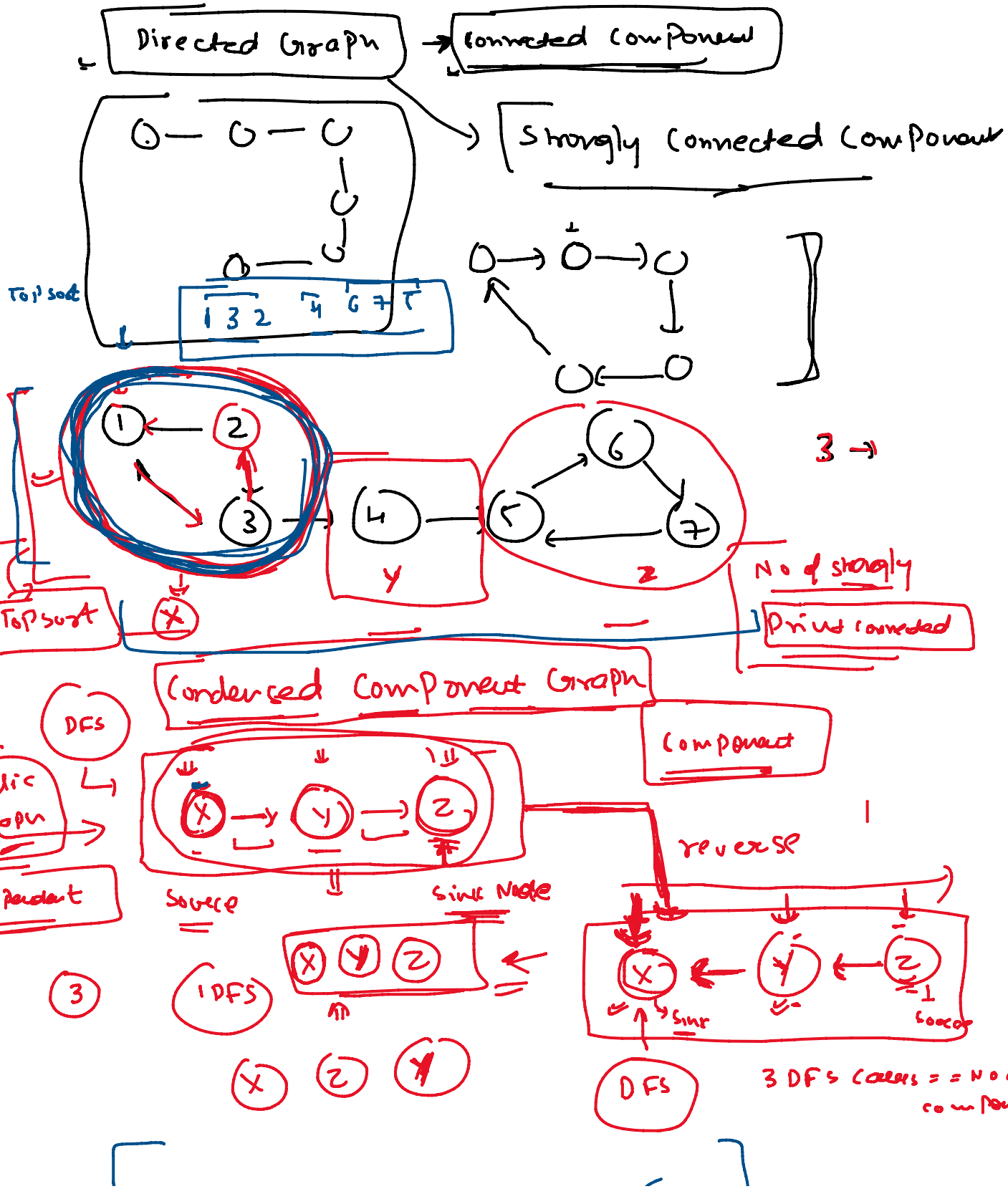
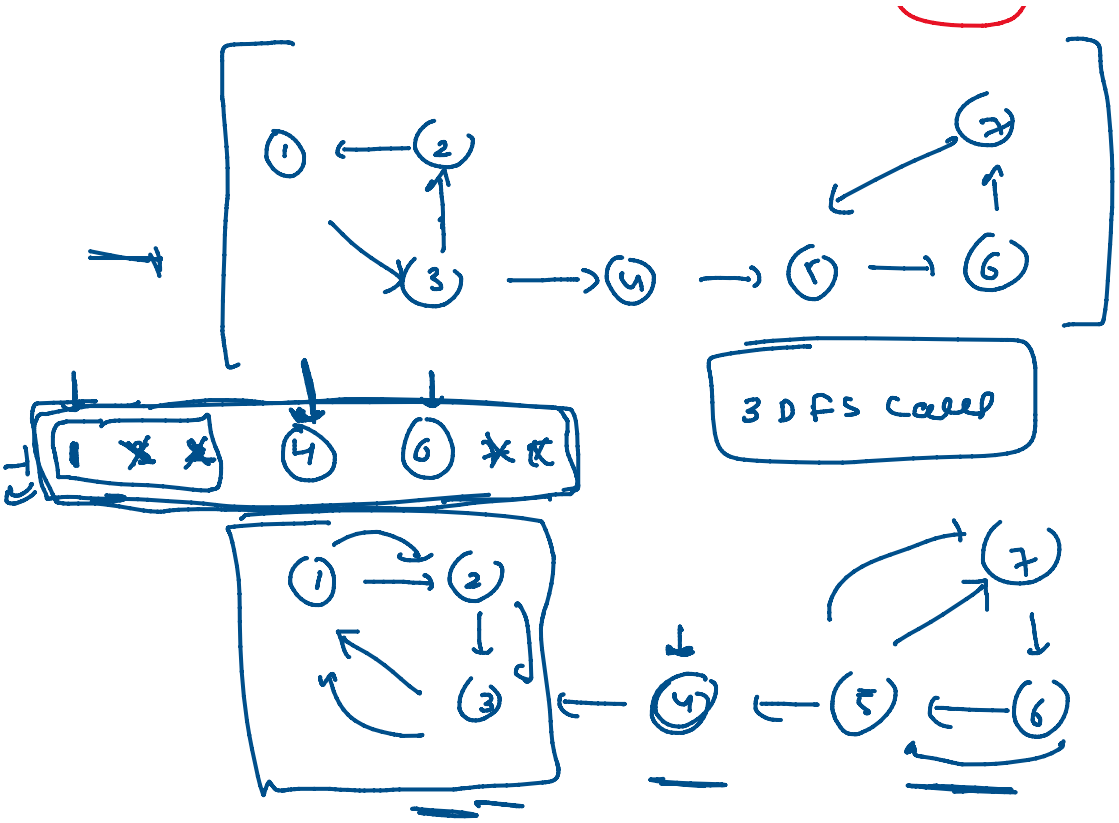
Class 101

02 June 2023 21:09

Kosaraju Algorithm

Connected Component





Code

```
vector<vector<int>> adj(n);

vector<int> top;

vector<int> vis(n+1,0);

for(i=1; i<=n; i++)
  if(!vis[i])
    dfs(i, top, adj, vis);

⇒ [reverse(top.begin(), top.end())]

vector<vector<int>> transpose(n+1);
```

```
void dfs(int node, vector<int>&
  top, vector<vector<int>>,
  vector<int>&vis)
{ vis[node]=1;
  for(auto i: adj[node])
  { if(!vis[i])
      { dfs(i, ...)
      }
  }
  top.push_back(node);
}
```

```
vector<vector<int>> transpose(n+1);
for( i=1 ; i<=n ; i++)
{   for(auto j : adj[i])
    {   transpose[j].push_back(i);
    }
}
```

i ← j

Reverse Graph

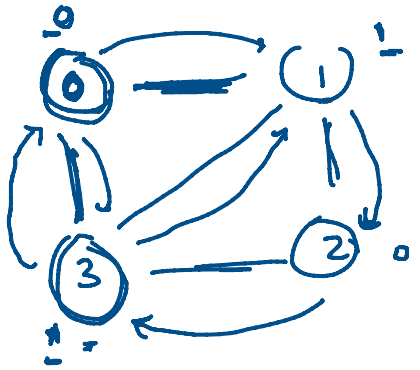
```
vector<int> vis2(n,0); int cnt=0;
for( i=0; i<top.size(); i++ )
{ if(!vis2[top[i])
  { dfs2( top[i], vis2, transpose )
    cnt++;
  }
}
```

cout << endl;

```
void dfs2( int node,
           vector<int>&vis,
           vector<vector<int>>&graph)
{ vis[node]=1;
  cout << node << " ";
for(auto i: transpose[node])
{ if(vis[i]==0)
    dfs2(i, vis, transpose);
}
```

⇒ Euler Path, Euler Circuit Hamiltonian Cycle =

0 , 1



Nodes Nodes

Bipartite

Bipartite

1

^ 1

0

0

```
bool bip(vector<vector<int>>&graph,int node,vector<int>&color,int col)
  {
      color[node]=col;

     for(auto i:graph[node])
     {
         if(color[i]!=-1)
         {
             if(color[i]==col)return false;
         }
         else
         {
             if(bip(graph,i,color,col^1))
             {
                 ;
             }
             else
                 return false;
         }
     }
     return true;

  }
   bool isBipartite(vector<vector<int>>& graph) {
      int i,j,k,n;
      n=graph.size();
      vector<int>color(n,-1);
      for(i=0;i<n;i++)
      {
          if(color[i]==-1)
          {
              bool ans=bip(graph,i,color,0);
              if(!ans)return false;
          }
      }
      return true;

   }
```

color(i) != -1

0 1

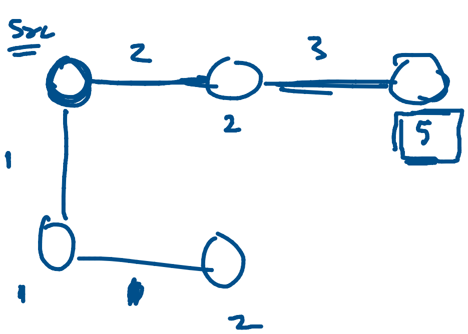
^ 1

1

0, 1

vis Array ← [

(-1) X



BFS

Space

Dijkstra → Set (4) ← 10 → 1 X

→ Min distance Node

Priority-queue → MinHeap

5 → 1

Priority_que → MinHeap

5 → 1

TLE

mindistance Node

(0,1)

(5,1)

[Carrier, Seaweed, Mo's Algo, Central Dream]

(CN), Sprinklr

#Uber