Graph Algorithms

CS3104

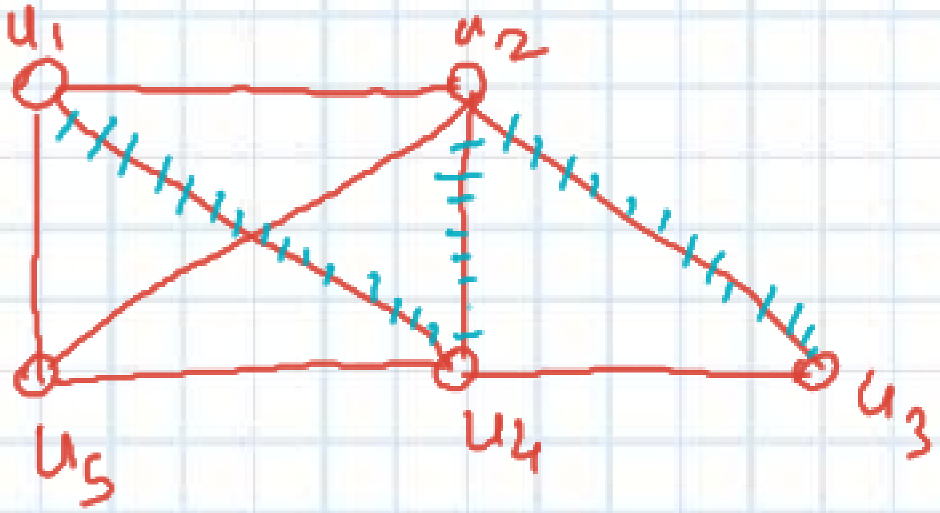
Dr. Samit Biswas, *Assistant Professor,*
Department of Computer Sc. and Technology,
Indian Institute of Engineering Science and Technology, Shibpur

Email: samit@cs.iiests.ac.in

Paths in Undirected Graph

- There is a *path* from vertex $\boldsymbol{v_0}$ to vertex $\boldsymbol{v_n}$ if there is a sequence of edges from v_0 to v_n
  - This path is labeled as $v_0, v_1, v_2, \ldots, v_n$ and has a length of n.
  - The path is a ***circuit*** if the path begins and ends with the **same vertex**.
  - A path is ***simple*** if it does not contain the same edge more than once.
  - A path or circuit is said to ***pass through*** the vertices $v_0, v_1, v_2, \ldots, v_n$ or *traverse* the edges $e_1, e_2, \ldots, e_n$.

Path - Example



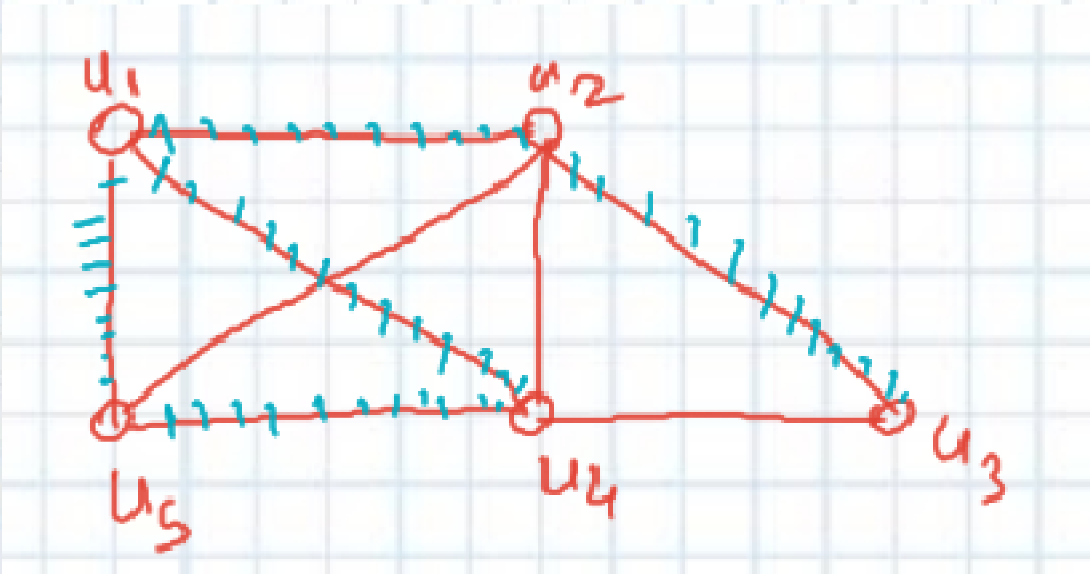
Consider this path
$u_1 \; u_4 \; u_2 \; u_3$

Simple

Length : 3

Circuit No

Path - Example



Path $u_1\ u_5\ u_4\ u_1\ u_2\ u_3$

length = 5

Circuit =

$u_1\ u_5\ u_4\ u_1$

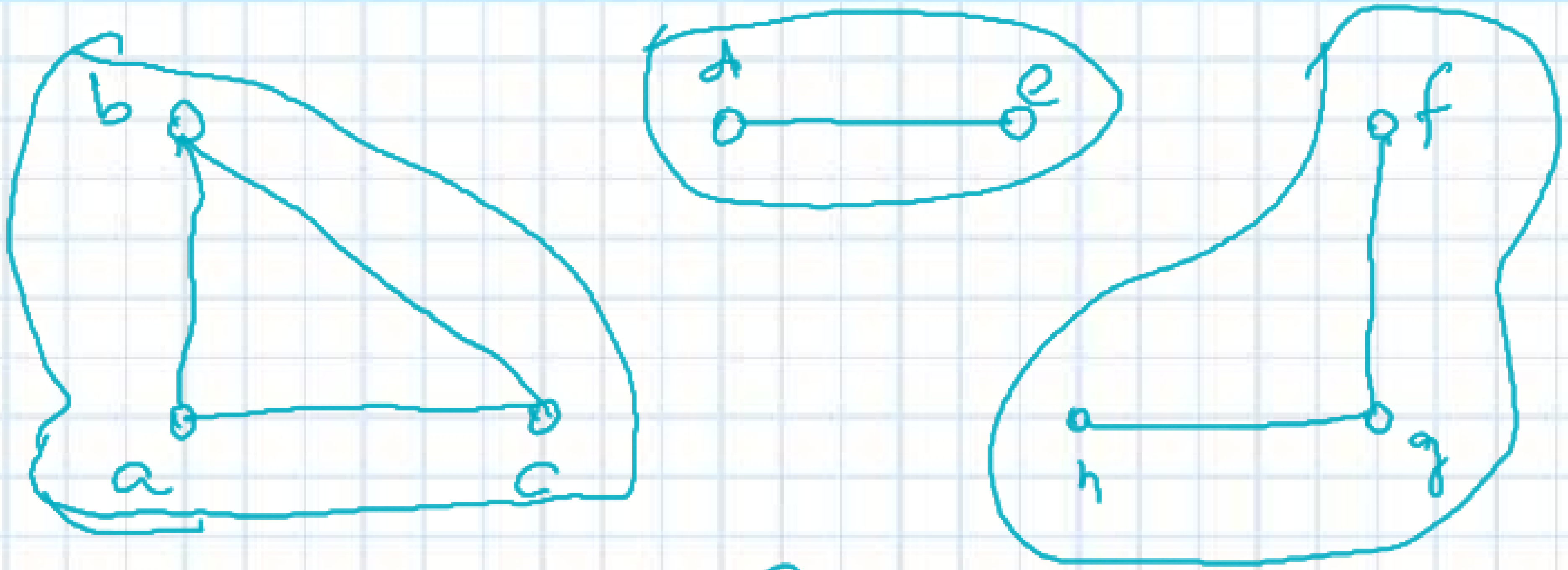
$u_1\ u_2\ u_5\ u_4\ u_3$

length = 4

Connectedness

- A graph that is not connected if the union of two or more disjoint connected subgraphs (called the *connected components* of the graph).

Connectedness - Example



G

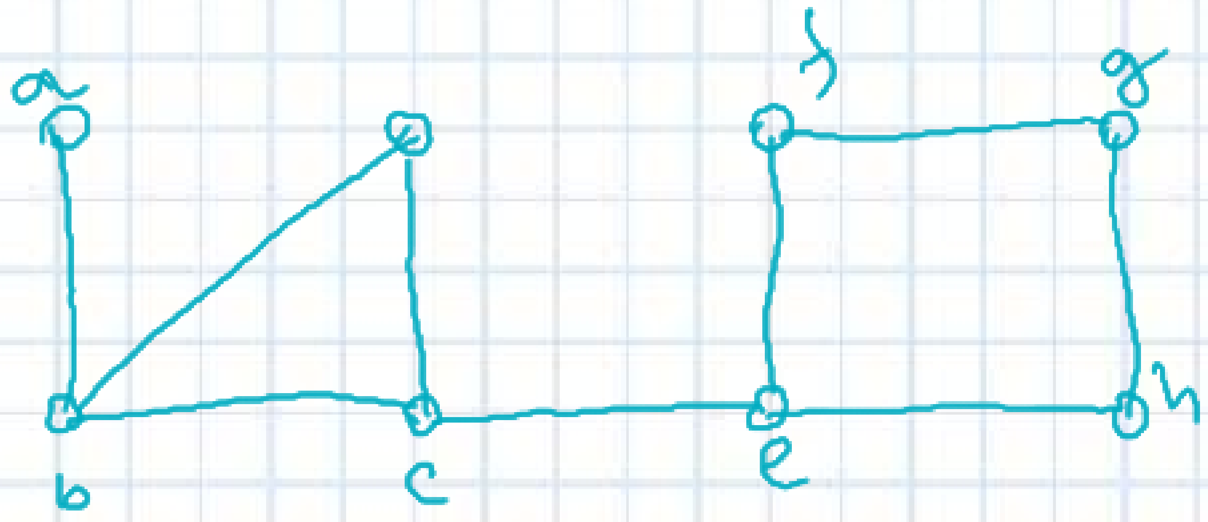
What are the connected components

$\{a, b, c\}$ $\{d, e\}$ $\{f, g, h\}$

Cut edges and vertices

- If one can remove a vertex (and all incident edges) and produce a graph with more connected components, the vertex is called a ***cut vertex***.
- If removal of an edge creates more connected components the edge is called a ***cut edge or bridge***.

Example



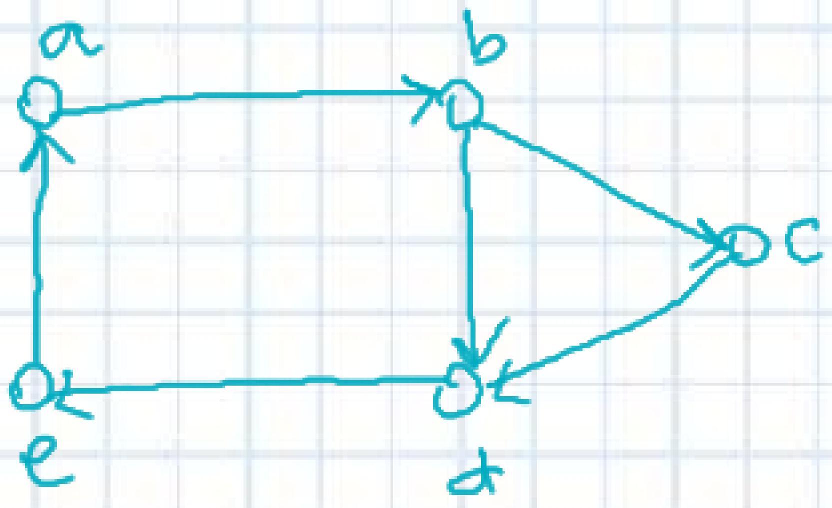
cut vertexs : c and e

cut edge : $\{(c, e)\}$

Connectedness in Directed Graph

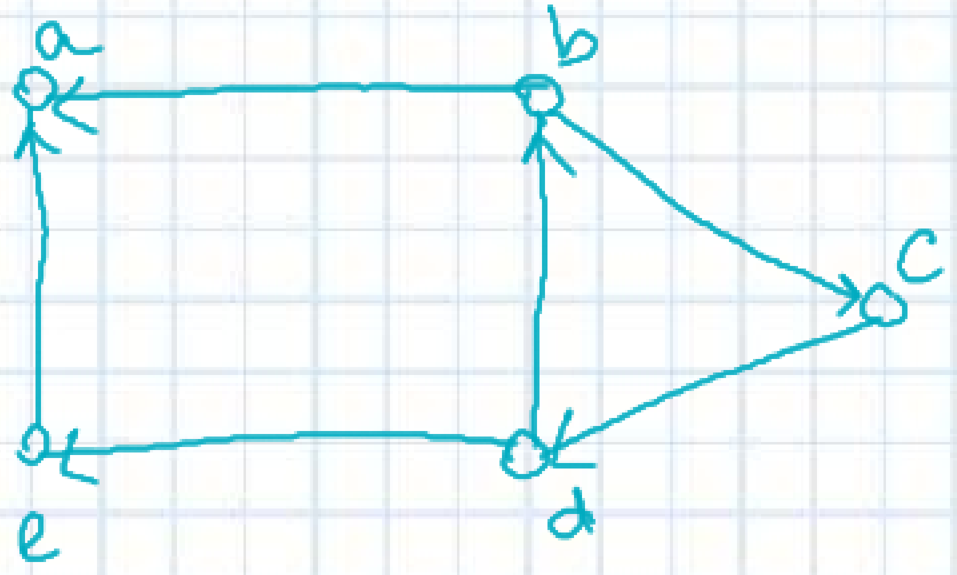
- A directed graph is *strongly connected* if there is a **directed** path between every pair of vertices.
- A directed graph is *weakly connected* if there is a path between every pair of vertices in the **underlying undirected** graph.
- The ***subgraphs*** of a directed graph G that are strongly connected but not contained in larger strongly connected subgraphs (the maximal strongly connected subgraphs) are called the **strongly connected components** or ***strong components*** of G.

Example



Strongly connected

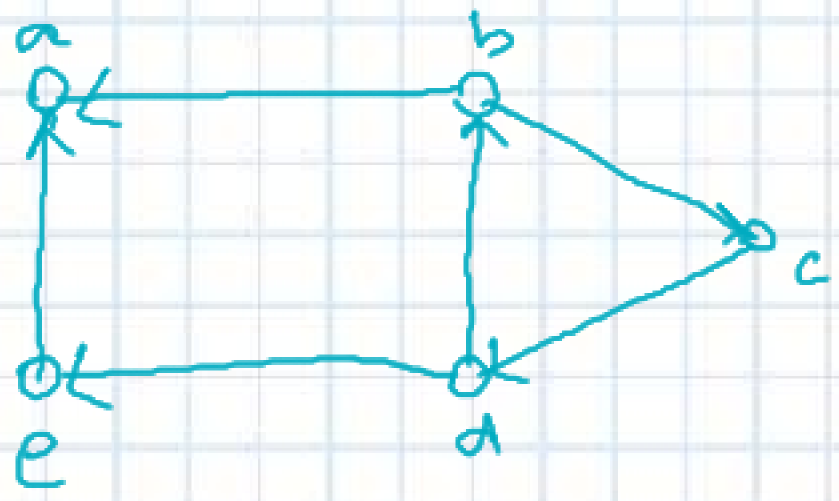
There is directed path bet^n every pair of vertices.



This graph is not strongly connected

$\longrightarrow$ no directed path bet^n $\boxed{\text{a and b}}$, $\boxed{\text{a and e}}$

Example



What are the strongly connected components?

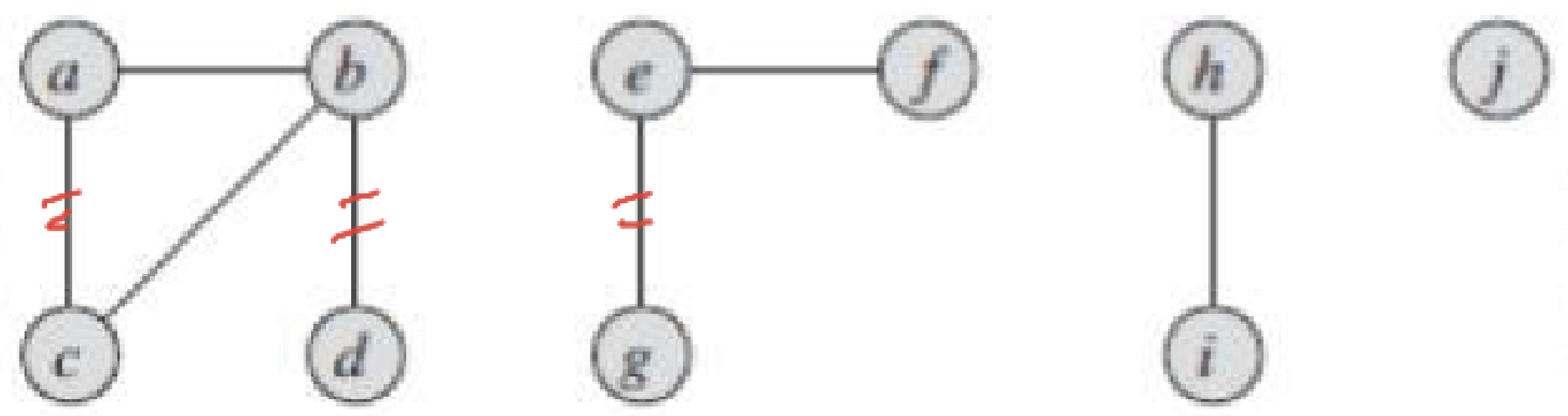
Components

$\{b, c, d\}$, $\{a\}$, $\{e\}$

Disjoint-Set Operations

- **MAKE-SET(x)**: creates a new set whose only member (and thus representative) is x.
- **UNION(x, y)**: unites the dynamic sets that contain x and y, say S_x and S_y, into a new set that is the union of these two sets. We assume that the two sets are disjoint prior to the operation.
- **FIND-SET(x)**: returns a pointer to the representative of the (unique) set containing x.

Example




```
CONNECTED-COMPONENTS(G)
for each vertex v ∈ G.V
    MAKE-SET(v)
for each edge (u, v) ∈ G.E
    if FIND-SET(u) ≠ FIND-SET(v)
        UNION(u, v)
```

Edge Processed	Collection of disjoint sets									
Initial sets	{a}	{b}	{c}	{d}	{e}	{f}	{g}	{h}	{i}	{j}
(b,d)	{a}	{b,d}	{c}		{e}	{f}	{g}	{h}	{i}	{j}
(e,g)	{a}	{b,d}	{c}		{e,g}	{f}		{h}	{i}	{j}
(a,c)	{a,c}	{b,d}	{c}		{e,g}	{f}		{h}	{i}	{j}
(h,i)	{a,c}	{b,d}			{e,g}	{f}		{h,i}		{j}
{a,b}	{a,b,c,d}				{e,g}	{f}		{h,i}		{j}
{e,f}	{a,b,c,d}				{e,f,g}			{h,i}		{j}

{a,b,c,d} {e, f, g}, {h, i} {j}