No affin lies today.

**Breadth First Search (BFS)**

last time: \( R \leftarrow \emptyset \)

\[
\text{while } \exists (u,v) \text{ s.t. } u \in R \land v \notin V \setminus R \text{ then }
\]

add \( v \) to \( R \) and
make \( v \) the child of \( u \)

\[\text{O/p the tree rooted at } s.\]

---

**Diagram:**

- A tree with nodes labeled from 'a' to 'h'.
- The root is 's', and branches extend to 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h'.
- Paths and edges indicate the structure of the tree.

---

\[\text{end of page}\]
$L_0$ contains $s$

$L_1$ contains all the neighbors of $s$.

$L_2$ contains all the neighbors of the neighbors of $s$ that have not yet been discovered.

$L_{l+1}$ contains all the vertices that are neighbors of vertices in $L_i$ that have not yet been discovered.

Lemma: Output of BFS is a tree (rooted at $s$)
**Lemma**: Suppose a vertex $v \in L_i$ in the BFS tree. Then $\text{dist}(s,v) = i$. Draw an edge from $s$ to $i$ in $G$.

**Lemma**: Let $T$ be the order of BFS. Let $u$ and $v$ be vertices in $G$ such that $(u,v) \in E$. Let $u \in L_i$ and $v \in L_j$ in $T$. Then $|i - j| \leq 1$.

**Proof Sketch**: Can I: $i = j$. Done.

Can II: $i \neq j$. WLOG, let $i < j$. Draw a diagram showing the relationships between $L_i$, $L_j$, $L_{i+1}$, and the vertices $u$, $v$, and $i$. The diagram illustrates the connectivity between these sets and vertices in the graph.
\[ j = i + 1. \]

**Graph Representation**

**Adjacency Matrix**

\[
M:
\begin{array}{cccc}
1 & 2 & j & n \\
1 & & & 0/1 \\
2 & & & \\
\vdots & & & \\
3 & & & \\
\end{array}
\]

- if \( (i,j) \in E \) then \( M[i,j] = 1 \)
- else \( M[i,j] = 0 \)

**Adjacency List**

all neighbors of 1.
0.1 \((u, v) \in E\) ?

\[ O(1) : \text{Adj. Matrix.} \]
\[ O(n) : \text{Adj list } (O(deg(u))) \]

0.2 Span Complexity?

\[ \Theta(n^2) : \text{Adj Matrix} \]
\[ \Theta(n+m) : \text{Adj list:} \]
\[ \Theta(n + \sum \text{dy}(u)) \]
\[ = \Theta(n + 2m) = \Theta(n+m). \]

0.3 Suppose we are at vertex \(u\) and we want to say "hi" to \(u\)'s neighbors.

\[ \text{Adj Matrix : } \Theta(n) \]
Adj list: \( \Theta(d_{\text{avg}}) \)

\[ \text{0(1) per neighbor.} \]

We will use Adjacency list.

BFS \((G, s)\)

disc.

\[
\begin{align*}
\text{for each } u \in V & \text{ do} \\
\text{discovered } (u) & \leftarrow \text{False} \\
\text{discovered } (s) & \leftarrow \text{True} \\
L[0] & \leftarrow \{s\} \\
T & \leftarrow \emptyset \\
\end{align*}
\]

\[
\begin{align*}
i & \leftarrow 0 \\
\text{while } L[i] \neq \emptyset & \text{ do} \\
L[i+1] & \leftarrow \emptyset \\
\text{for each } u \in L[i] & \text{ do} \\
\text{for each } v \in N(u) & \text{ do} \\
\text{discovered } (u) & \leftarrow \text{False} \\
\end{align*}
\]

implemented using a Queue
\[
\begin{align*}
&\text{if } \text{discovered}[v] = \text{false} \\
&\quad \{\text{Enqueue } (u,v) \text{ to } T \\
&\quad \quad \text{discovered}[v] \leftarrow \text{true} \\
&\quad \text{let } i \leftarrow i + 1 \\
&\text{return } T
\end{align*}
\]

Running time:

- Each vertex in \( G \) appears in a \( L[C] \) list at most once.
- For each vertex in a layer, we go to each of its neighbors & say "hi".
  \( O(dy(u)) \) time.

: while loop takes

\[
O\left( \sum_v dy(v) \right) = O(n)
\]

: Running time of the algorithm: \( O(n+m) \).
\( O(n^2) \rightarrow \) pessimistic estimate when \( G \) is sparse.

\( O(n+m) \rightarrow \) pruned.

\[ O/P: \]

Suppose we want a connected component; we go to a vertex that is not yet discovered & start a new tree from there.

**Testing Bipartiteness.**

**Input:** Undirected graph \( G = (V,E) \)

**Obj.** Output Yes, if \( G \) is bipartite

No, o.w.

Bipartite graphs \( \xrightarrow{2\text{-colorable}} \)

\( \xrightarrow{\text{Check each edge \((u,v)\) & o.w.}} \)
\[ \overline{V} \]

- no odd-length cycles. 

**Alg.**

1. \( T \leftarrow \text{BFS}(G) \)

2. Color all even layers **Blue** & all odd layers **Red**.

3. for each \( e = (u,v) \) in \( G \) do
   
   if \( u \) & \( v \) are colored the same then
   
   o/p \( \text{No} \)

4. o/p \( \text{Yes} \).

**Correctness**

Can I: Our alg. returns \( \text{Yes} \).

\( G \) is indeed bipartite.
We want to argue that "God" cannot color us if 2 colors.

Let (u,v) be an edge in G s.t.

u & v are colored the same (this is where our alg o/p is NO).

Case I: u & v belong to diff layers.
Cannot happen because of the Lemma.

\(\text{Case II: } u \& v \text{ belong to the same layer.}\)

Odd length cycle formed using \(u, v \& s\) lower common ancestor of \(u \& v\). (1st)

\[\text{dist}(s, u) = \text{dist}(s, v)\]

\[\therefore \text{length of the cycle } = 2 \cdot \text{dist}(s, u) + 1\]

Is \(G\) 3-colorable?

\[O(n^{1000})\]