Learning Goals

- Technically: Distinguish when BFS vs. Dijkstra applies; How to implement Dijkstra.
- Civically: See how 'weights' encode *whose* costs or barriers matter in real life (influence, trust, access, language, bandwidth, safety).

Graph Review

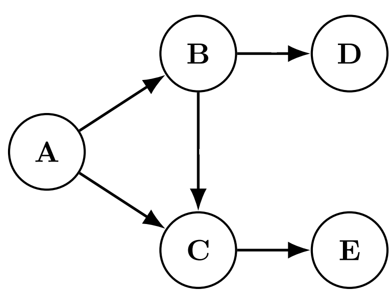
1. An **unweighted** graph treats every edge with the same ____________________.
2. A **weighted** graph assigns each edge a ____________________ (time, cost, risk, etc.).
3. A **directed** graph has edges with ____________________ (one-way links).
4. An **undirected** graph has edges usable in ____________________ directions.

Shortest Path Problem in Social Network

In a social network or communication system, the shortest path represents the most efficient route for information to travel between people or groups.

A. BFS and Unweighted Graph

Here, nodes represent 5 people online; edges indicate ways they connect and share information. A post starts at node **A**. Level refers to the depth or distance of a node from the starting node.



Task. Using BFS from source **A**, list the first time each node 'sees' the post (fewest shares). Fill discovery order and parent pointers:

Vertex v	Ava	Bob	Chloe	Dylan	Emma
Level (steps from A)					
BFS parent	–				

Shortest (fewest shares) path from A to E: ____

Civic Reflection (discuss in groups) In this *equal–step* model, who gets the post earliest and why? Who might be disadvantaged if some connections do not exist (rural users, language communities, people off major platforms)?

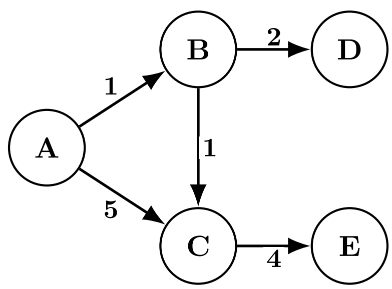
Give one concrete real-world advantage of using BFS here, and one risk of assuming all edges are equally easy to use.

- Advantage: ____
- Risk: ____

Bridge Idea. BFS assumes *equality of access*: every edge costs '1 share' or "1 call." Real life adds frictions: language, bandwidth, attention, safety, trust, time, money.

Part B. Dijkstra and Weighted 'Influence/Trust' Graph

Now we add edge weights to encode influence / trust / effort. Weight represents difficulty for information to pass through that direction. A larger weight means it is *harder* or *costlier* for information to travel that edge (e.g., low trust, language barrier, limited data plan). Smaller weight = easier spread. With the same 5 people, imagine Ava is posting to defend against a rumor or raise awareness about issues.



Task Run **Dijkstra** on your own, from source **A**. Fill a standard table each time you extract the next vertex u and relax outgoing edges.

Dijkstra Trace Table (fill during run):

Step	Extracted u	$d[A]$	$d[B]$	$d[C]$	$d[D]$	$d[E]$	Checked edges / Notes
Init	–	0	∞	∞	∞	∞	Initialize all $d[\cdot]$
1							
2							
3							
4							
5							

Shortest weighted path (min total effort) from A to E: ______________________________

Civic Reflection (discuss in groups). Compared to BFS, which route changed and why? What does your weight choice implicitly value (e.g., high-trust ties, high-bandwidth users, dominant language groups)? Who benefits/hurts?

Part C.Pseudocode

BFS(G, start) (for unweighted graphs)

```
for each vertex v in G:



distance[start] =
create an empty queue Q
enqueue(Q, start)

while Q is not empty:
    u = dequeue(Q)
    for each neighbor v of u:





            enqueue(Q, v)
```

Dijkstra(G, start) (for weighted graphs with no negative weights)

```
for each vertex v in G:



distance[start] =
create a priority queue PQ (ordered by smallest distance)
insert all vertices into PQ

while PQ is not empty:
    u = vertex in PQ with smallest distance
    remove u from PQ
    for each neighbor v of u:





            update PQ with new distance[v]
```

Part D. Compare & Reflect

1. When would you intentionally choose **BFS** for social systems (equal steps)? Give one concrete case where that equality is desirable.

2. When must you use **Dijkstra** because equality hides inequity? Name one weight you would add for a fairer outreach or safer information flow.

Meta-Reflection:
When all connections are treated equally, we assume that every message spreads with the same ease—this is what Breadth-First Search (BFS) models.

However, in reality, connections often carry different "qualities", such as time delays, language barriers, or varying levels of trust. Using Dijkstra's algorithm and edge weights lets us represent these frictions and study how they affect who receives information first. By analyzing the shortest paths in a weighted graph, we can better understand inequalities in communication, reveal which links make a network more inclusive or exclusive, and design fairer ways for information to reach everyone.