Let's begin at 9:05 PM

L86
Multi-Source BFS, 0-1 BFS and more

Join Discord - https://bit.ly/ly-discord

RECAP

Quick Question

Can BFS be applied on a directed graph also?

A big fat YES!

Multi-Source BFS

Instead of putting a single source to the queue:

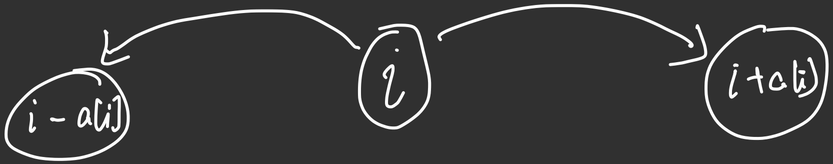
1) Put all of the sources to the queue
2) Make d[] value for all of them ⇒ 0

Introduction

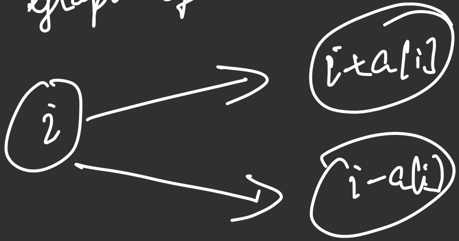
```
for ( int src : sources) {
      q.push(src)
      d[src] = 0;
}
while (!q. empty() {
     // Exactly same as BFS
 }
```

Practice Problem
Nearest Opposite Parity

Intuition / Solution



graph of N nodes



---

index i ⇒ bfs(i)

get the min. distance from the d array out of all the opp. parity nodes ⇒ $O(N^2)$

Actual Quest:->

Min. moves required to reach an opp. parity node from index i.

What if ?

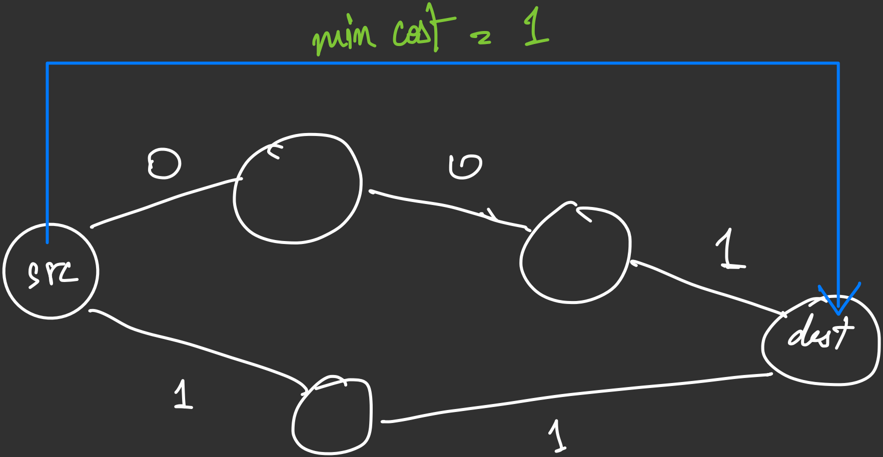
Min. moves required to reach the index i from any of the opp. parity nodes even nodes ⟹ Multi-source BFS [d Even]

odd nodes ⟹ Multi-source BFS again [d Odd]

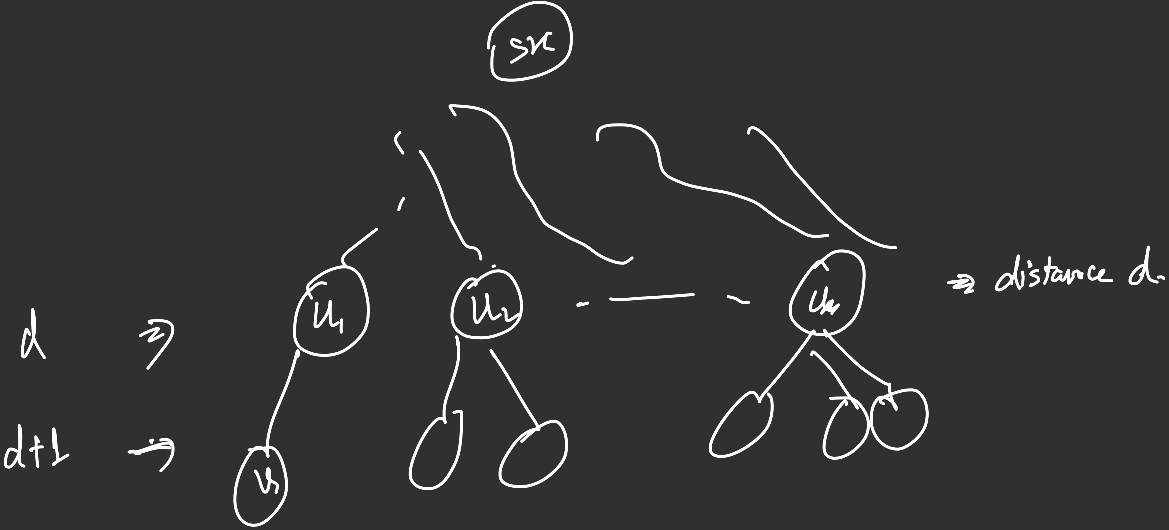
```
for (i=1; i<=N; ++i)
    if (a[i] % 2 == 0)
        ans[i] = dOdd[i];
    else
        ans[i] = dEven[i];
```

Let’s Implement

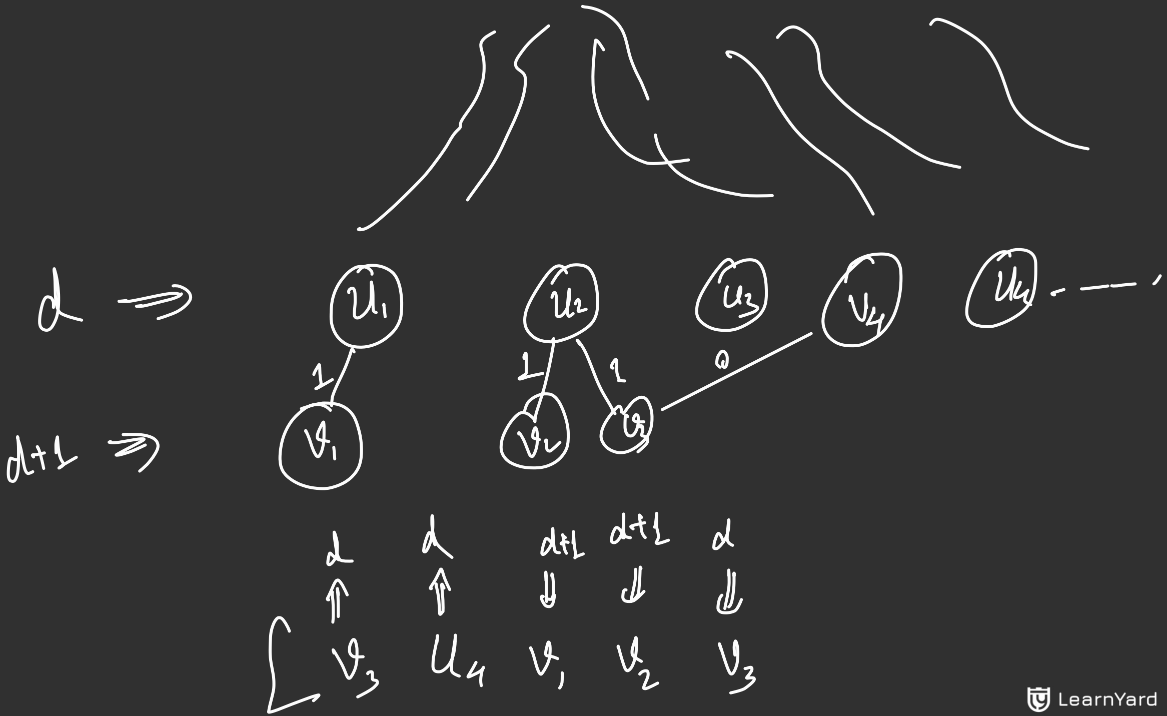
0-1 BFS



Introduction



$q \Rightarrow [v_{1d}, v_{2d} \cdots\cdots v_{1d+1}, v_{2d+1} \cdots]$


d
d+1
u1
u2
u3
v4
u4
v1
v2
v3
1
1
1
0
d
d
d+1
d+1
d
v3
U4
v1
v2
v3

d ⟹ inf

```
dNew = d[cur] + $w_e$

if (dNew < d[nb]) {
    q.add(nb);
    d[nb] = dNew;
}
```

⟹ if We is 1 back

⟹ 0 front

Thank You!

Reminder: Going to the gym & observing the trainer work out can help you know the right technique, but you’ll muscle up only if you lift some weights yourself.

So, PRACTICE, PRACTICE, PRACTICE!