

Shortest Distance Algorithms: (Single Source)

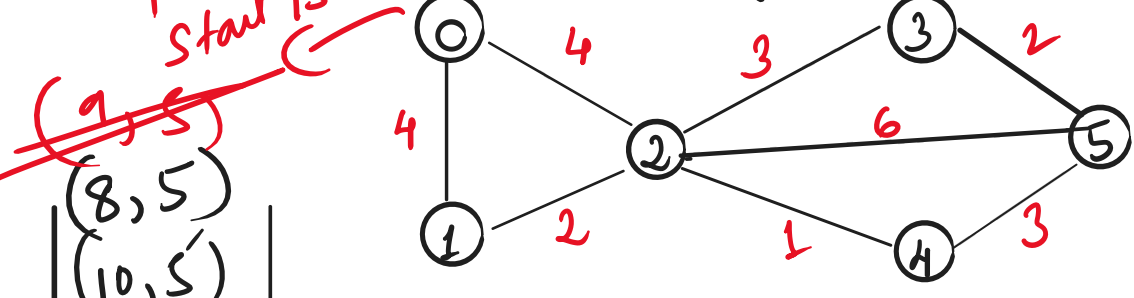
Dijkstra's Algorithm: →

Find the shortest distance from a source node to all other nodes in the graph.

Adjacency List: {Node, Weight}

- 0 → {1, 4}, {2, 4}
- 1 → {0, 4}, {2, 2}
- 2 → {0, 4}, {1, 2}, {3, 3}, {4, 1}, {5, 6}
- 3 → {2, 3}, {5, 2}
- 4 → {2, 1}, {5, 3}
- 5 → {2, 6}, {3, 2}, {4, 3}

Bjarne Stroustrup



$E \log V$

(8, 5)
(10, 5)
(5, 4)
(7, 3)
(4, 2)
(4, 1)
(0, 0)

priority queue
min heap

(dist, node)
distance array: →

0	1	2	3	4	5
0	4	4	7	5	8

Dynamic Programming: →

"Those who forget the past, are forced to repeat it."

* The solution of smaller sub-problems to solve a bigger problem. [Overlapping Subproblems]

* Recursion

Array { * Memoisation → when we don't calculate previous values.

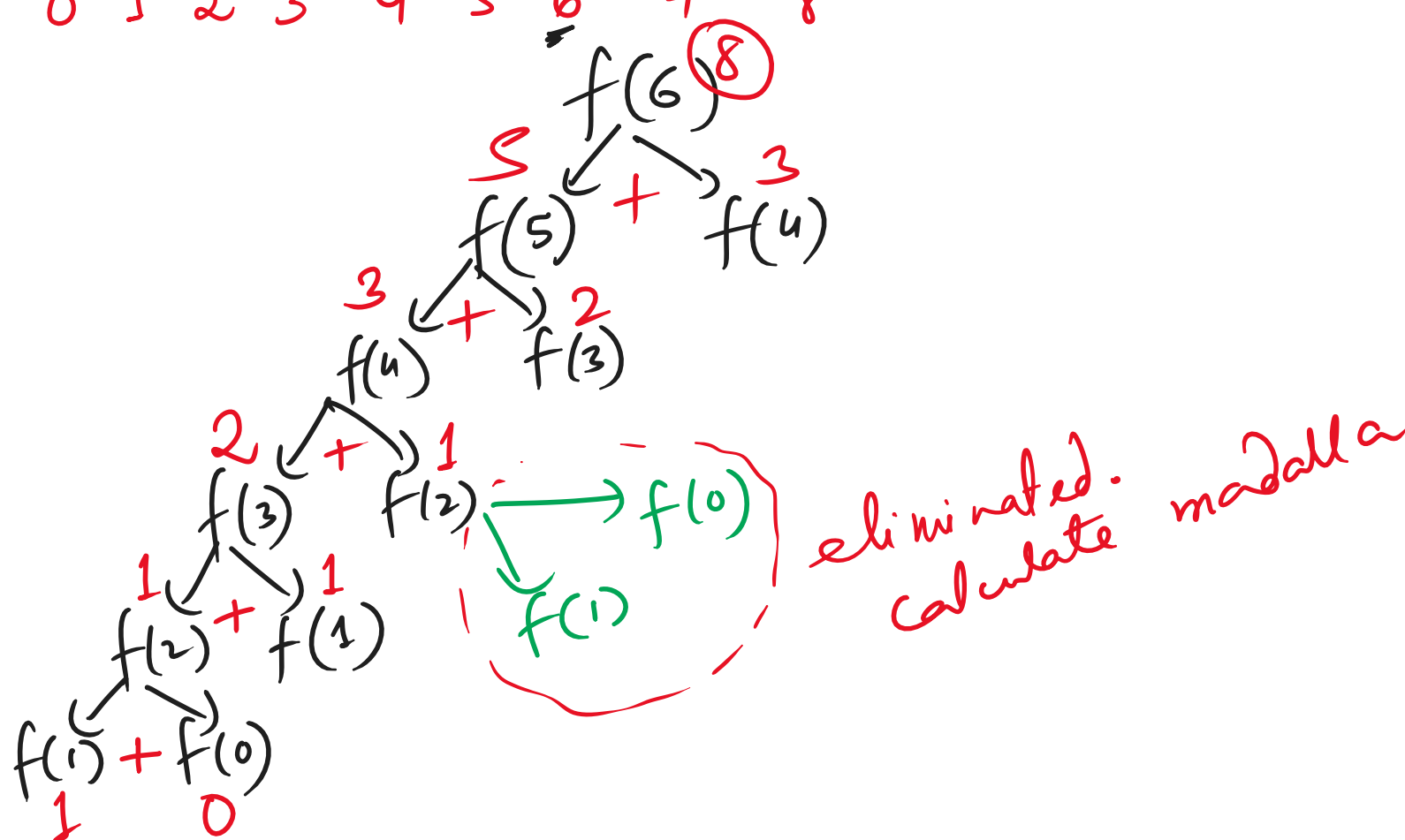
* Tabulation → store prev values.

* Space Optimization (Not possible for all)

Recursion Tree for Fibonacci Series: →

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

$$f(n) = f(n-1) + f(n-2)$$



0, 1, 1, 2, 3, 5, 8, 13, ...

p2 p1 curr

p2 p1 curr

$O(n)$

p2 = 0

p1 = 1

(return p1)

$O(1)$

only

curr = p2 + p1

(n)

3

1 = 0 + 1 = 1

variables

p2 = p1

p1 = curr

1

[Longest Common Subsequence]

Top Down

→ Abdul Bari

Bottom Up

(50k)

[Minimum number of steps]

Coders

C++ → Striver

→ take u forward

Aracade

Love Babbar → code help

Java → Kunal Kushwaha

5 days x 30hrs → DSA

[Feedback: 12180]