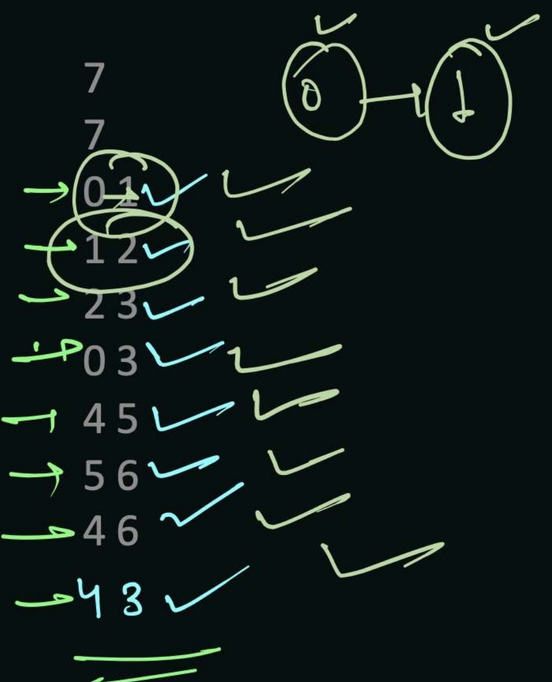
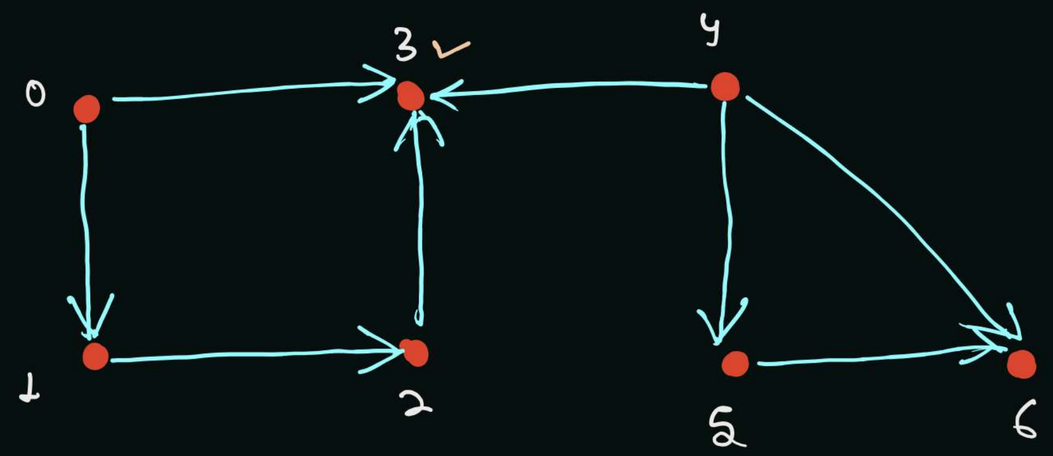
Order of Compilation / Topological Sort

Saturday, 10 July 2021 9:56 AM

Directed Acyclic Graph (DAG)





order of compilation

↓

Reverse

↓

topological Sort

order of comp.

0 - 3
1 - 6
2 - 5
3 - 4
4 - 2
5 - 1
6 - 0

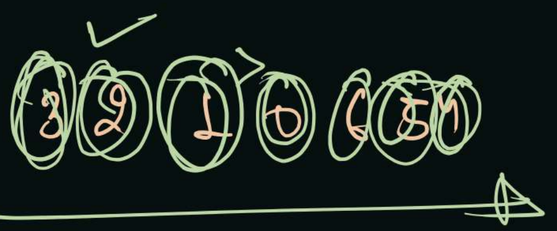
order of compilation →

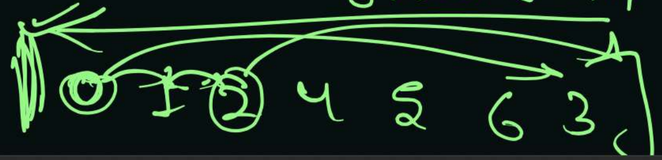
Vertices → function

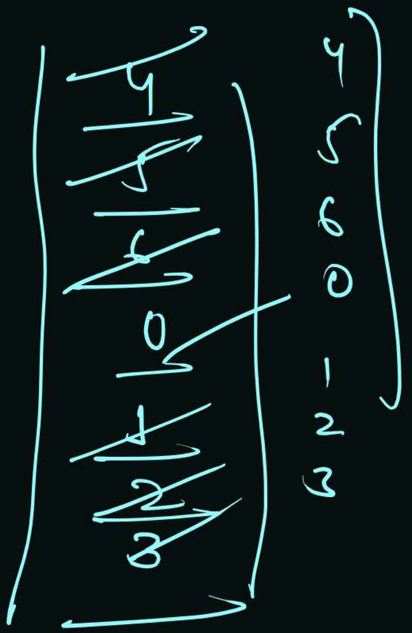
Edges → Dependencies

order of compilation

3
2
1
0
6
5
4







Order of compilation ?

a ✓
b ✓
c ✓
d ✓
e ✓
f ✓
g ✓
h ✓
i ✓
j ✓

marked

order of compilation →

e d b c a h g f j i

reverse of topological sort

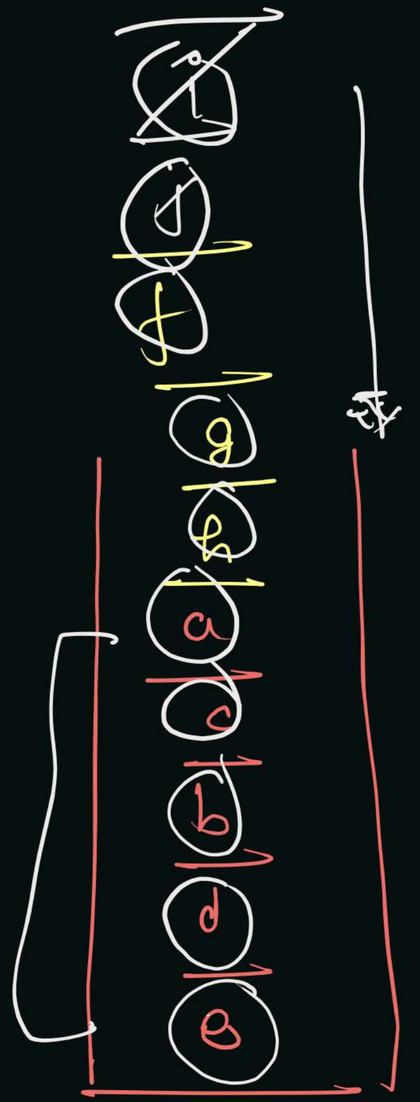
topological sort →

i j f g h a c b d e

Stack →

why pre Order printing] X] post order → add in stack

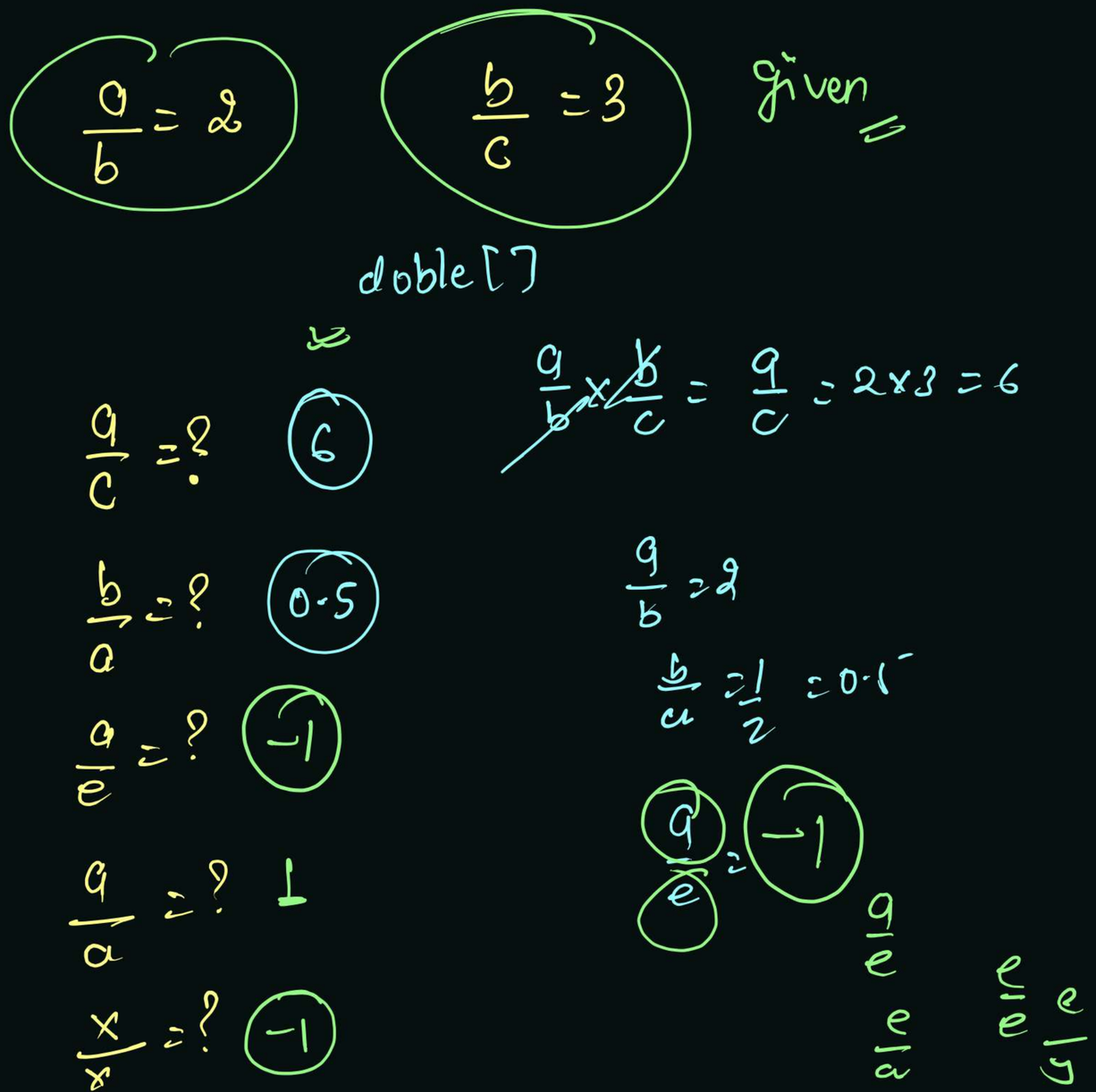
array - stack



Evaluate Division

Sunday, 11 July 2021 8:50 AM

Link : https://leetcode.com/problems/evaluate-division/



399. Evaluate Division

Medium 3676 287 Add to List Share

You are given an array of variable pairs `equations` and an array of real numbers `values`, where `equations[i] = [Ai, Bi]` and `values[i]` represent the equation `Ai / Bi = values[i]`. Each `Ai` or `Bi` is a string that represents a single variable.

You are also given some `queries`, where `queries[j] = [Cj, Dj]` represents the `jth` query where you must find the answer for `Cj / Dj = ?`.

Return *the answers to all queries*. If a single answer cannot be determined, return `-1.0`.

Note: The input is always valid. You may assume that evaluating the queries will not result in division by zero and that there is no contradiction.

Example 1:

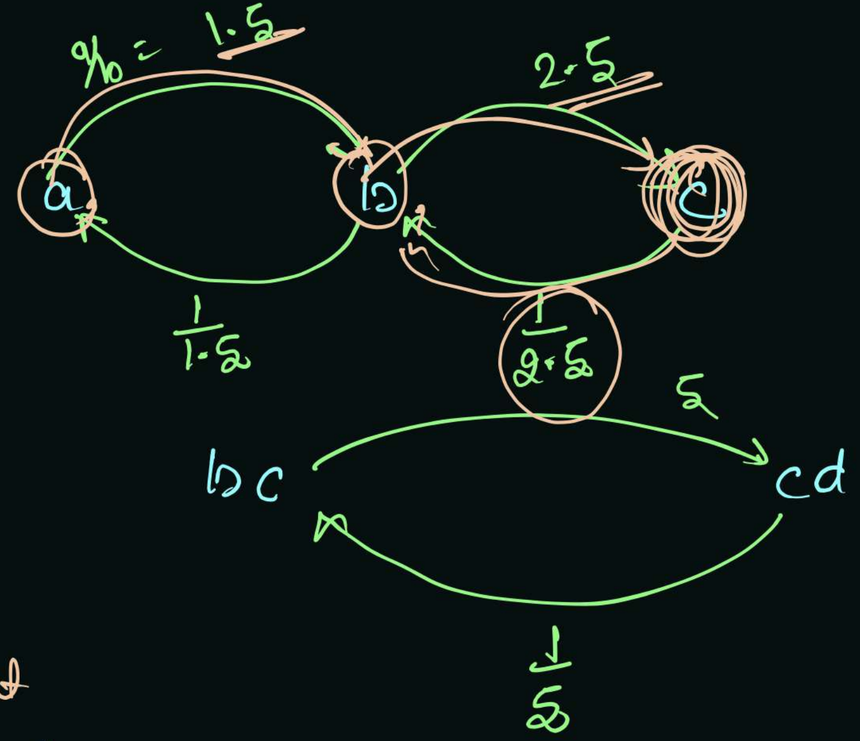
```
Input: equations = [["a","b"],["b","c"]], values = [2.0,3.0], queries = [["a","c"],["b","a"],["a","e"],["a","a"],["x","x"]]
Output: [6.00000,0.50000,-1.00000,1.00000,-1.00000]
Explanation:
Given: a / b = 2.0, b / c = 3.0
queries are: a / c = ?, b / a = ?, a / e = ?, a / a = ?, x / x = ?
return: [6.0, 0.5, -1.0, 1.0, -1.0 ]
```

Example 2:

these are Edges.

```
Input: equations = [["a","b"],["b","c"],["bc","cd"]], values =
[1.5,2.5,5.0], queries = [["a","c"],["c","b"],["bc","cd"],["cd","bc"]]
Output: [3.75000,0.40000,5.00000,0.20000]
```

weight



Absent = -1

Same ↓ different final undesired path = (1)

$\frac{a}{c} \rightarrow 1.5 \times 2.5$

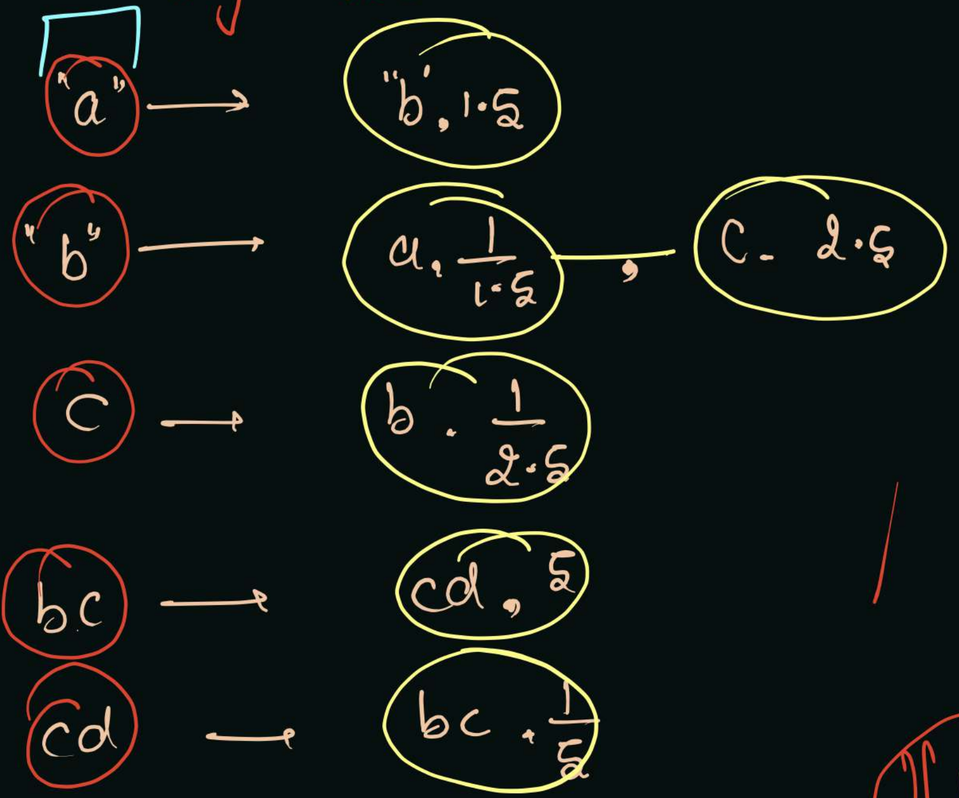
$\frac{a}{b} \times \frac{b}{c} = \frac{a}{c} = 1.5 \times 2.5$

$\frac{c}{b} \rightarrow$

$\frac{bc}{cd}$

Absent = -1 $\rightarrow \frac{cd}{bc}$

Example 2:



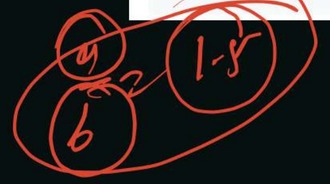
```
Input: equations = [["a","b"],["b","c"],["bc","cd"]], values =
[1.5,2.5,5.0], queries = [["a","c"],["c","b"],["bc","cd"],["cd","bc"]]
Output: [3.75000,0.40000,5.00000,0.20000]
```

Equation

wt

src

nbr



a

b

1.5

HashMap.

~~array~~

Key. Value.

"a" → "b", 1.5

"b" → a, $\frac{1}{1.5}$, c. 2.5

c → b. $\frac{1}{2.5}$

bc → cd, 5

cd → bc. $\frac{1}{5}$

HashMap <String, Arraylist<Edge>>

graph

Hashset → Key

get contains move → O(1)

Keyset