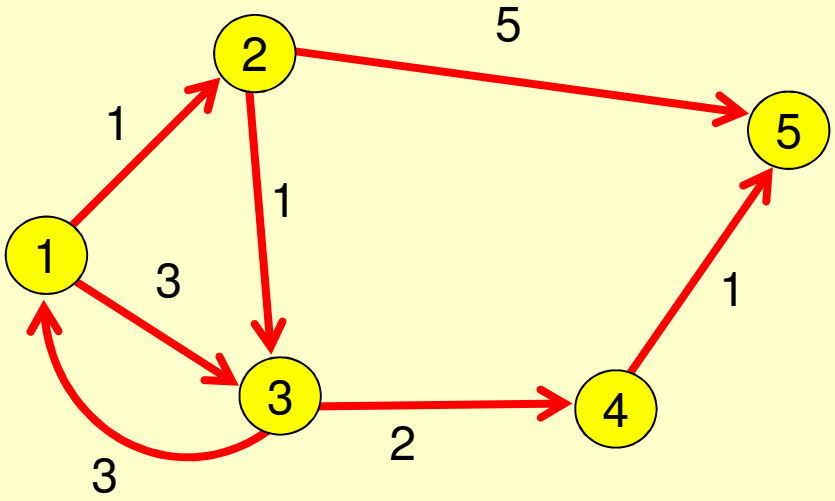
# Pembahasan Flowchart Shortest Path Problem

Ali Ridho Barakbah

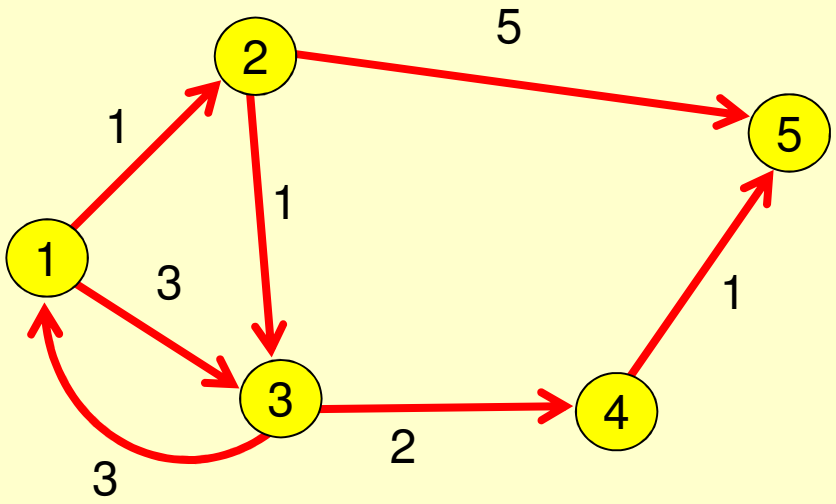
# Shortest Path Problem




Buatlah flowchart untuk menghitung jarak minimal dan rutenya dari titik 1 ke titik 5

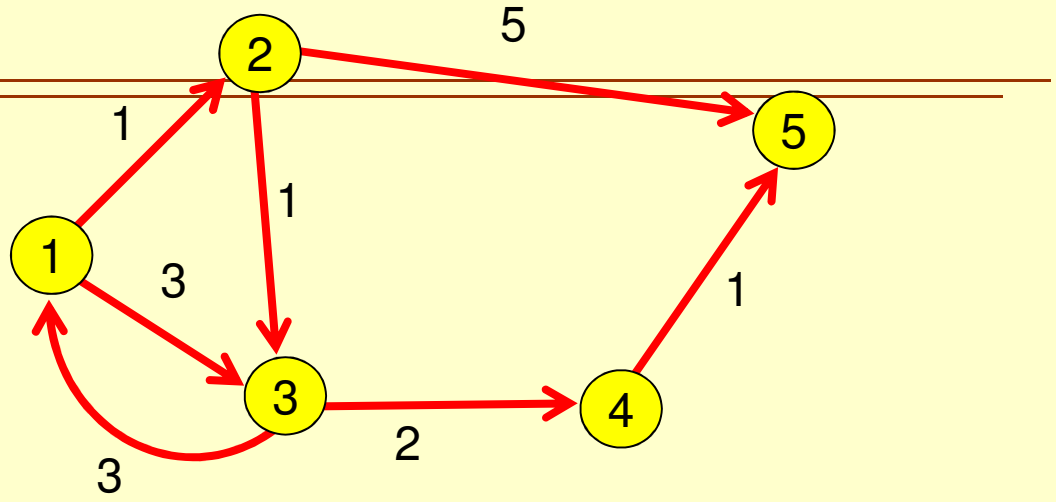
# Shortest path problem untuk Single path

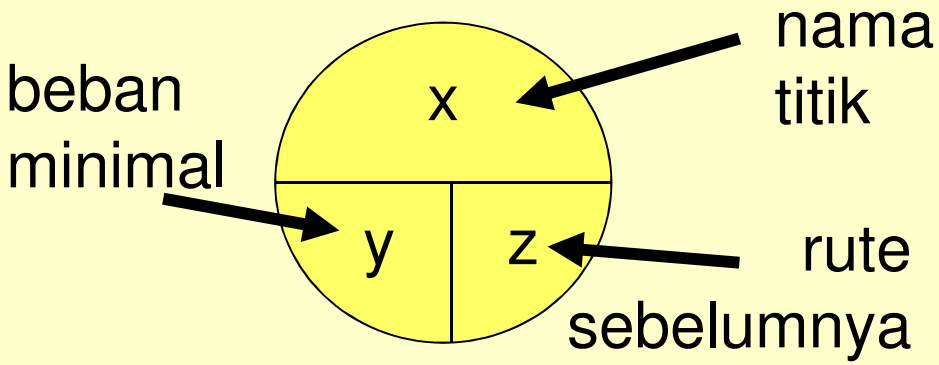
## (Algoritma Dijkstra)



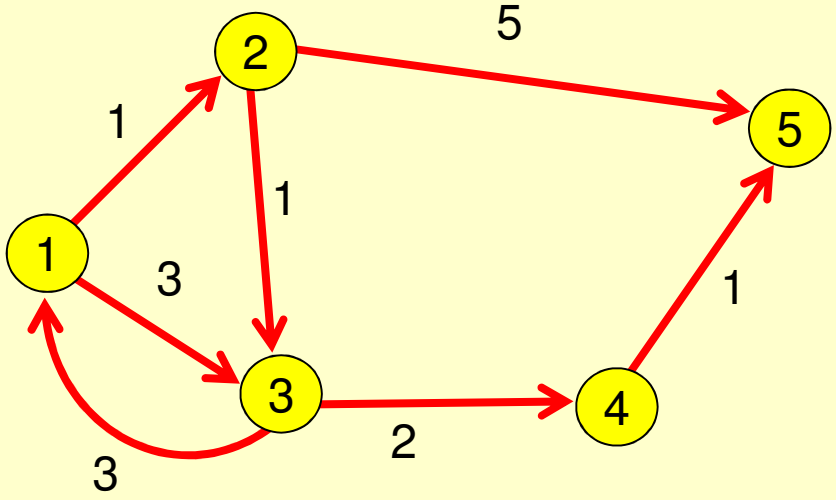

- Tentukan titik asal dan titik tujuan sebelum proses
- Akumulasikan jarak minimal dan simpan ke titik berikutnya. Lakukan dari titik asal sampai titik tujuan

Titik asal = 1
Titik tujuan = 5



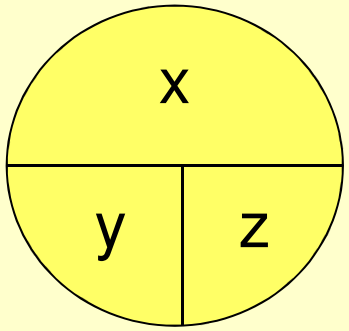



# Pemakaian Variabel




n ← jumlah titik
asal=1
tujuan=5

| relasi | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 1 |
| 3 | 1 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | 0 | 1 |
| 5 | 0 | 0 | 0 | 0 | 0 |

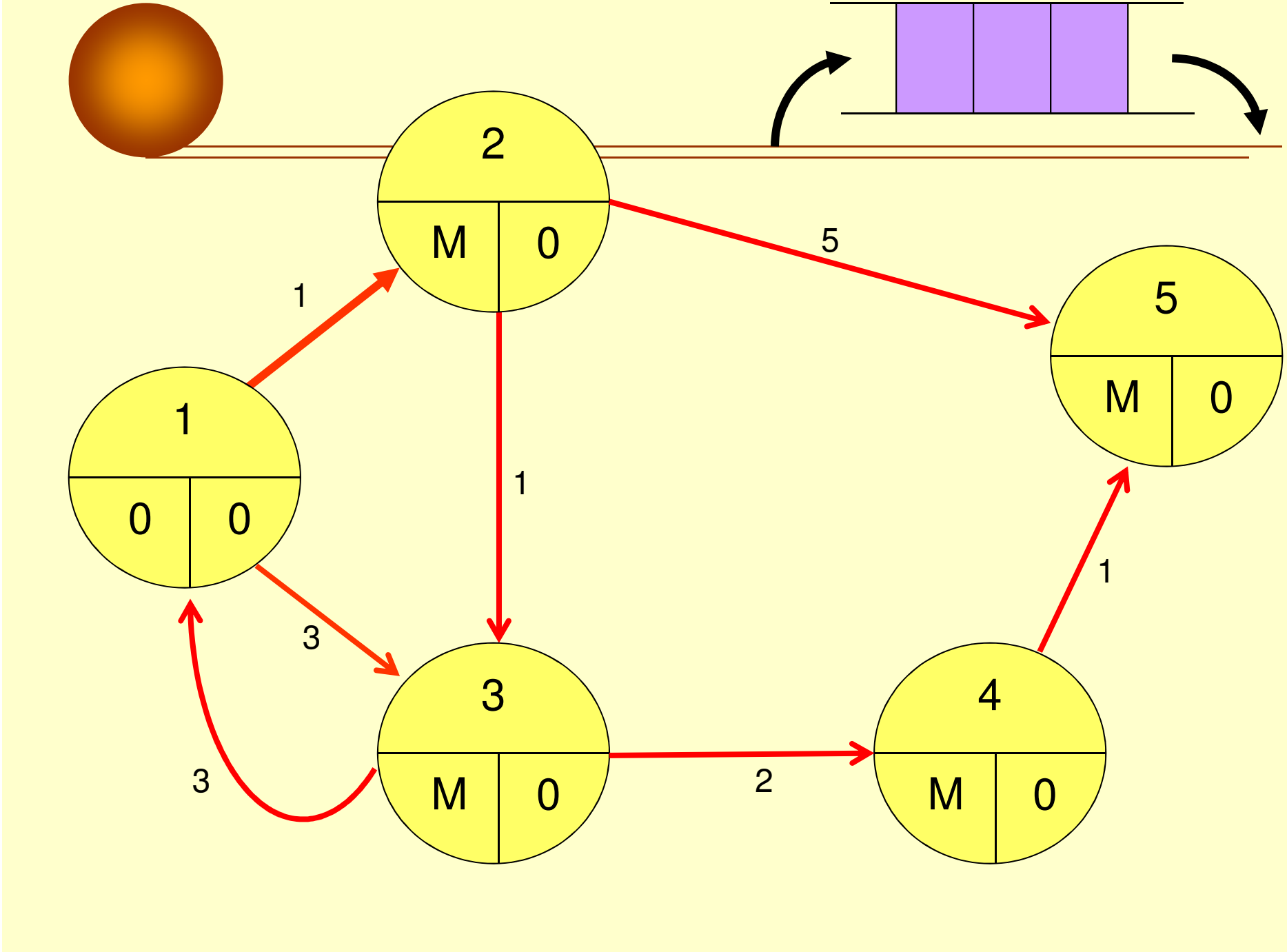


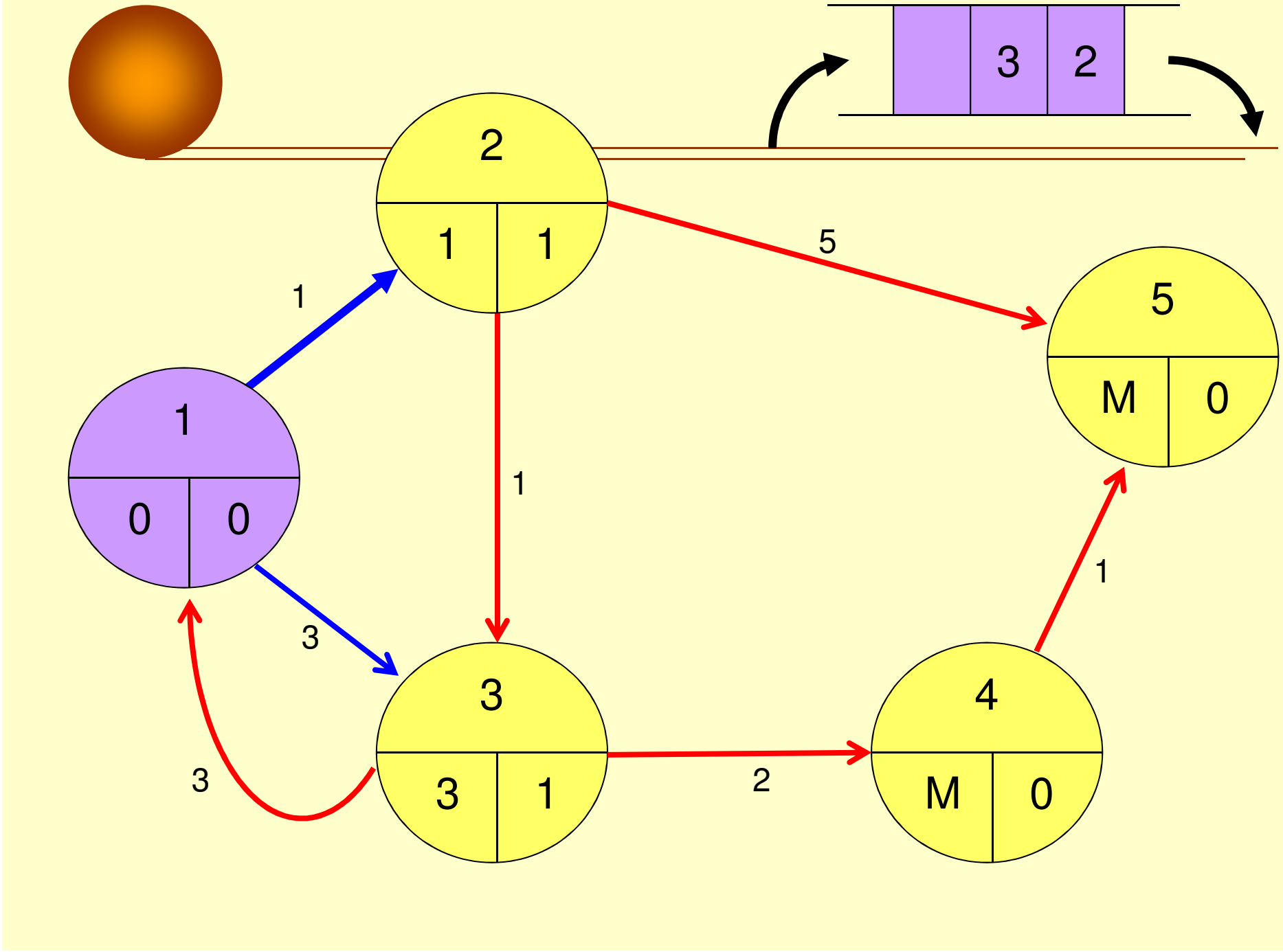

| | y |
|---|---|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |

| | z |
|---|---|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |

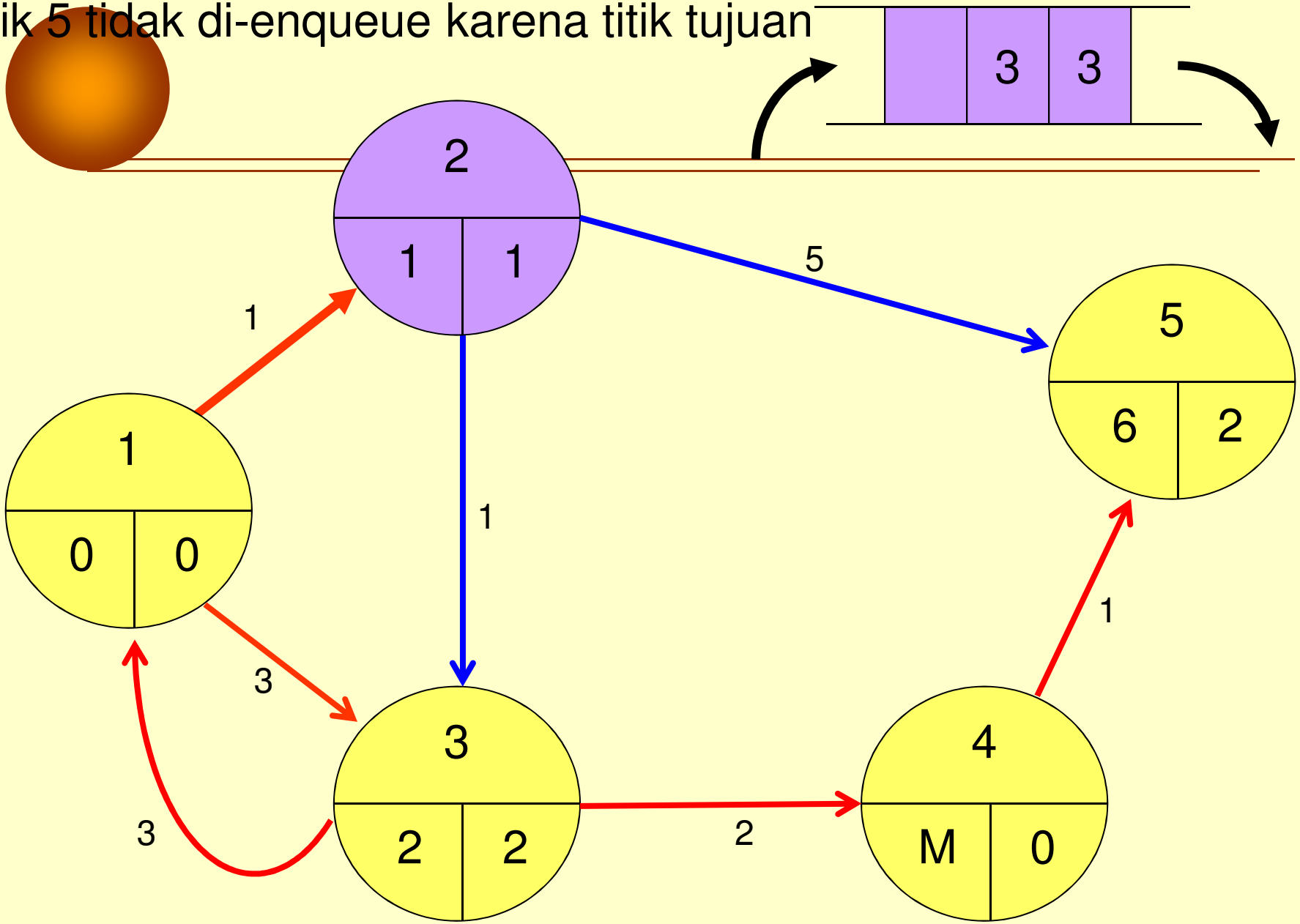
| jarak | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 3 | M | M |
| 2 | M | 0 | 1 | M | 5 |
| 3 | 3 | M | 0 | 2 | M |
| 4 | M | M | M | 0 | 1 |
| 5 | M | M | M | M | 0 |

M = 1000 (nilai yang besar)

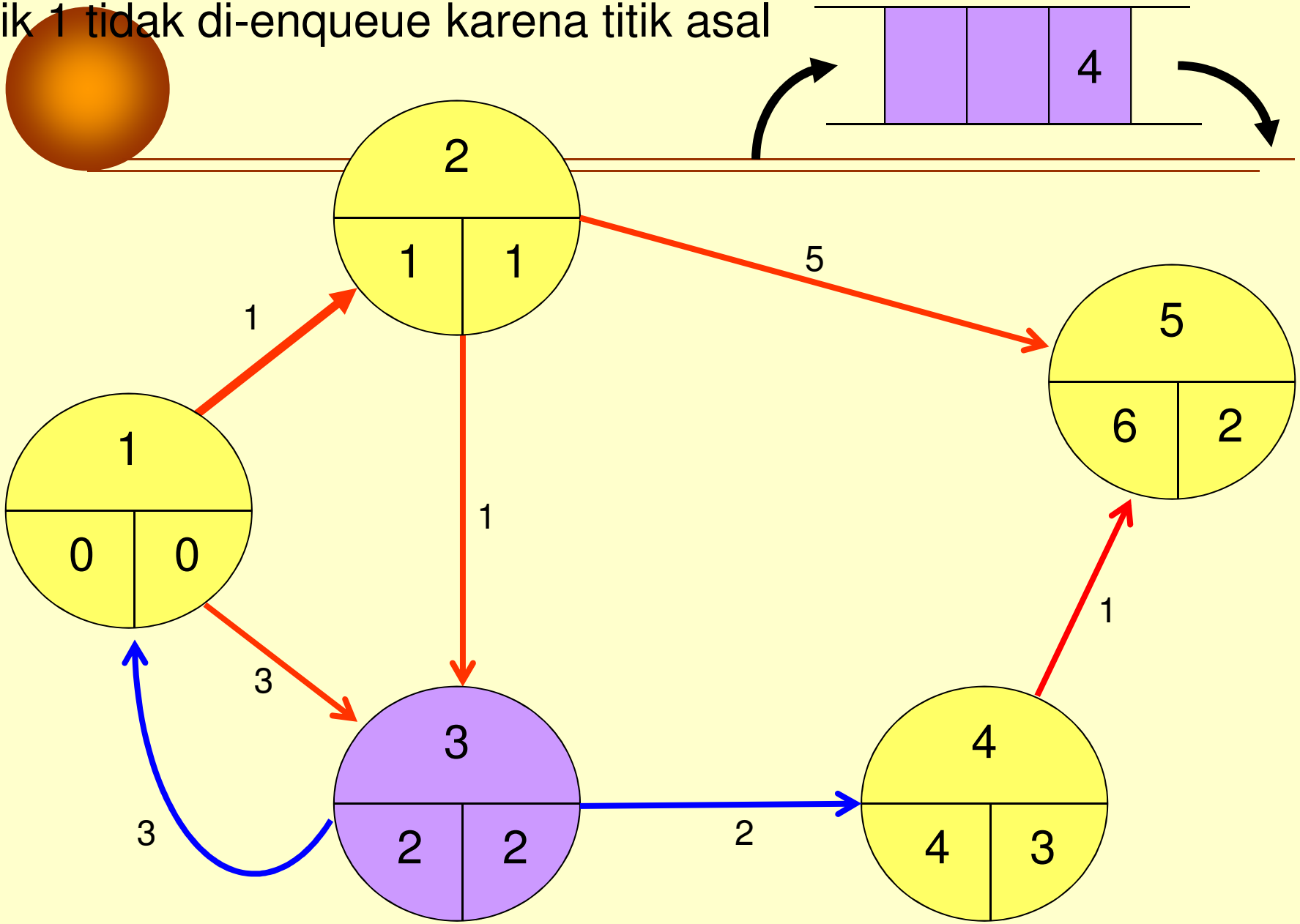

2
M
0
1
0
0
5
M
0
3
M
0
4
M
0
1
5
1
3
3
2
1


3
2
2
1
1
5
1
5
M
0
1
0
0
1
3
3
3
1
3
2
4
M
0
1

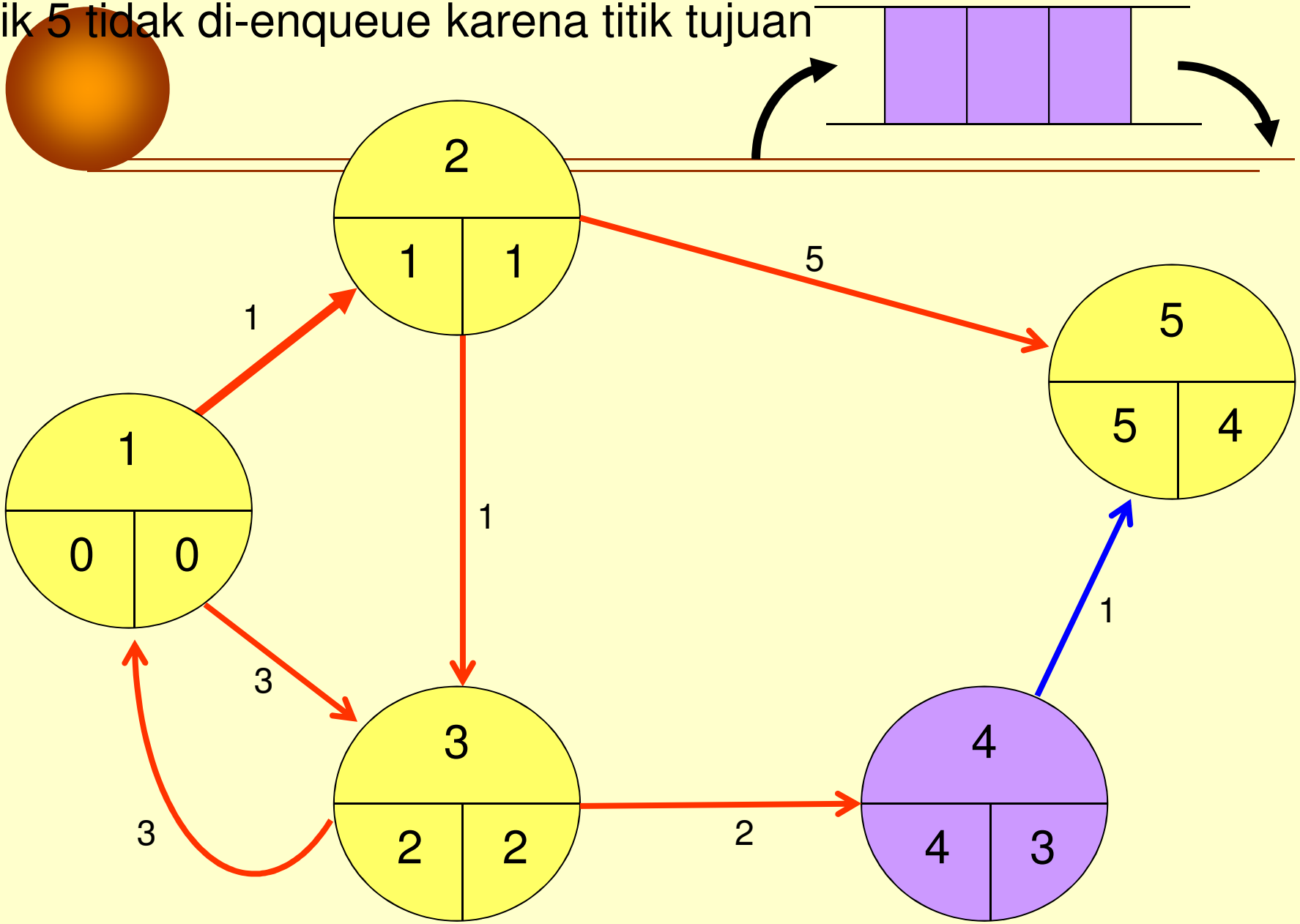
Titik 5 tidak di-enqueue karena titik tujuan

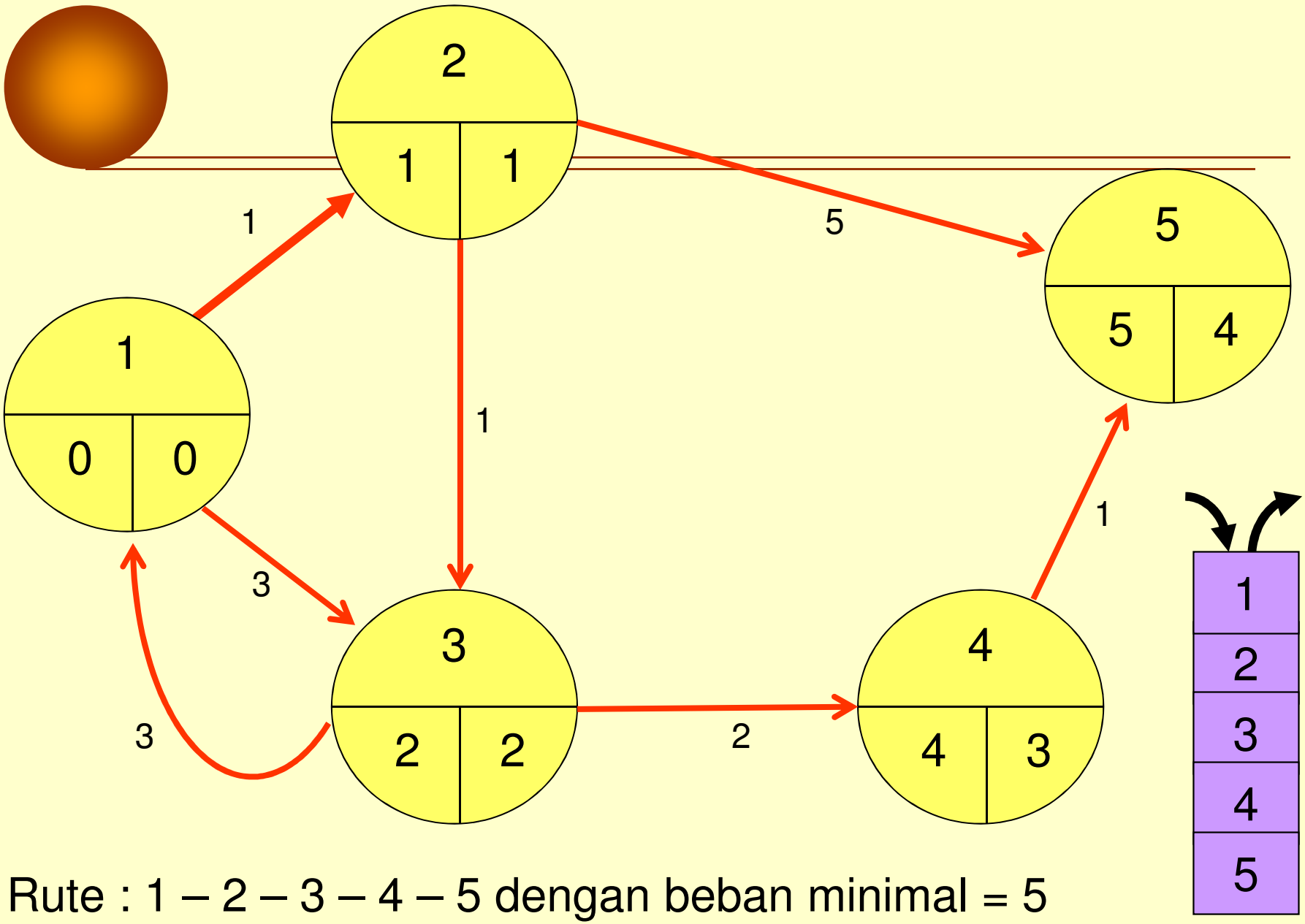


Titik 1 tidak di-enqueue karena titik asal

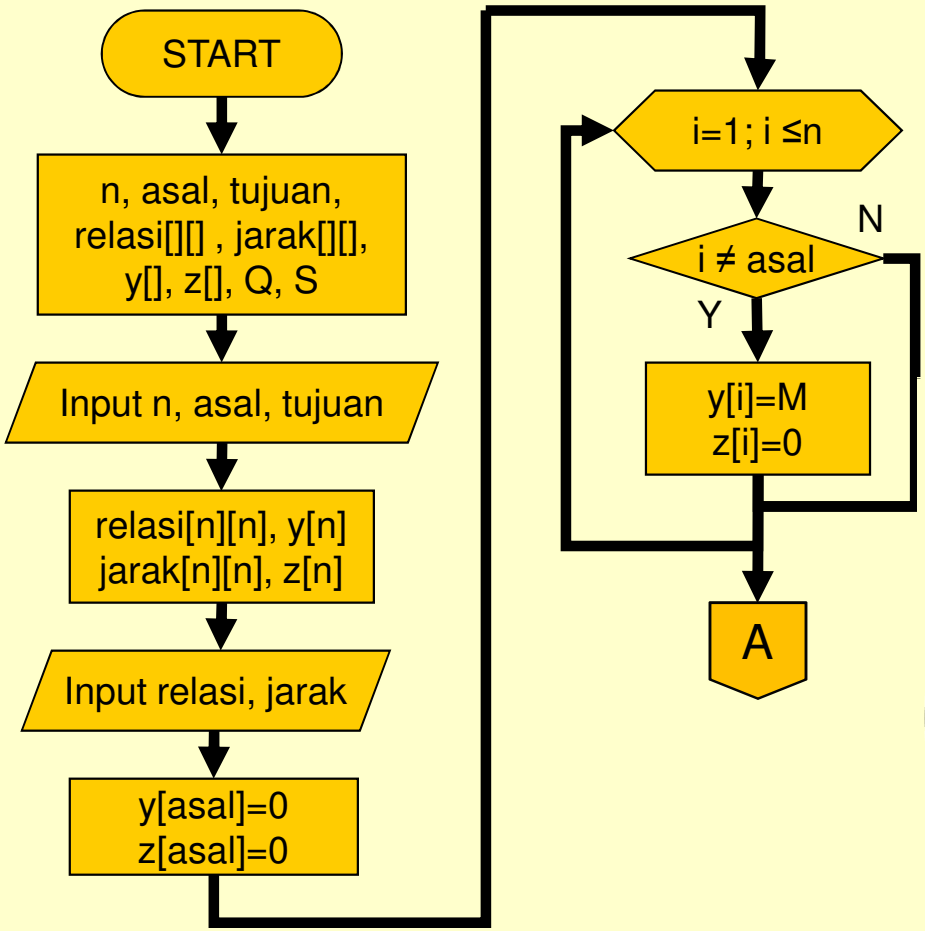


Titik 5 tidak di-enqueue karena titik tujuan

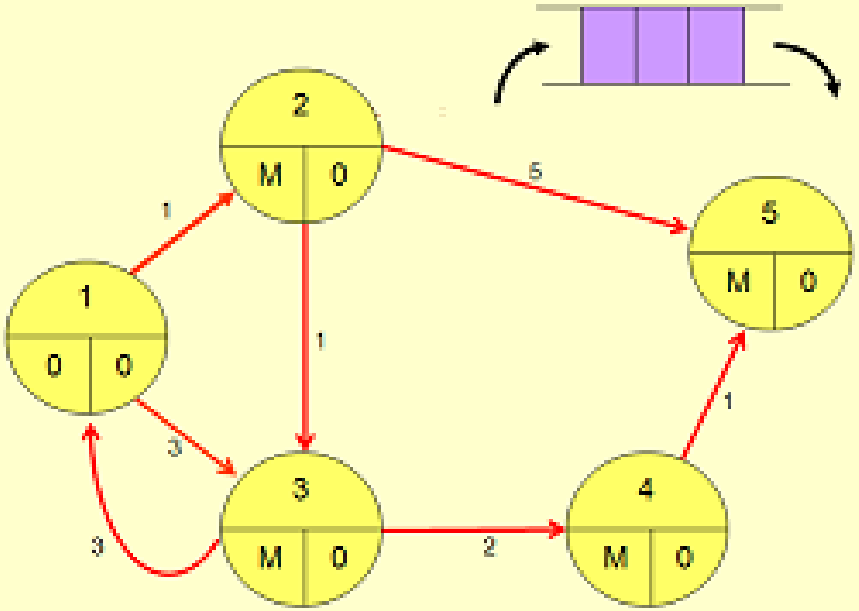


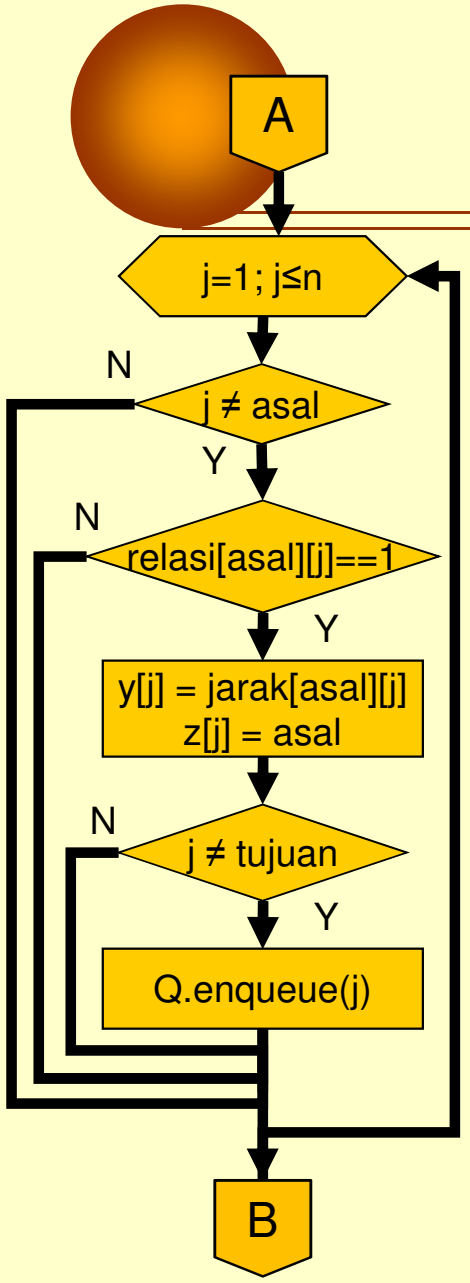


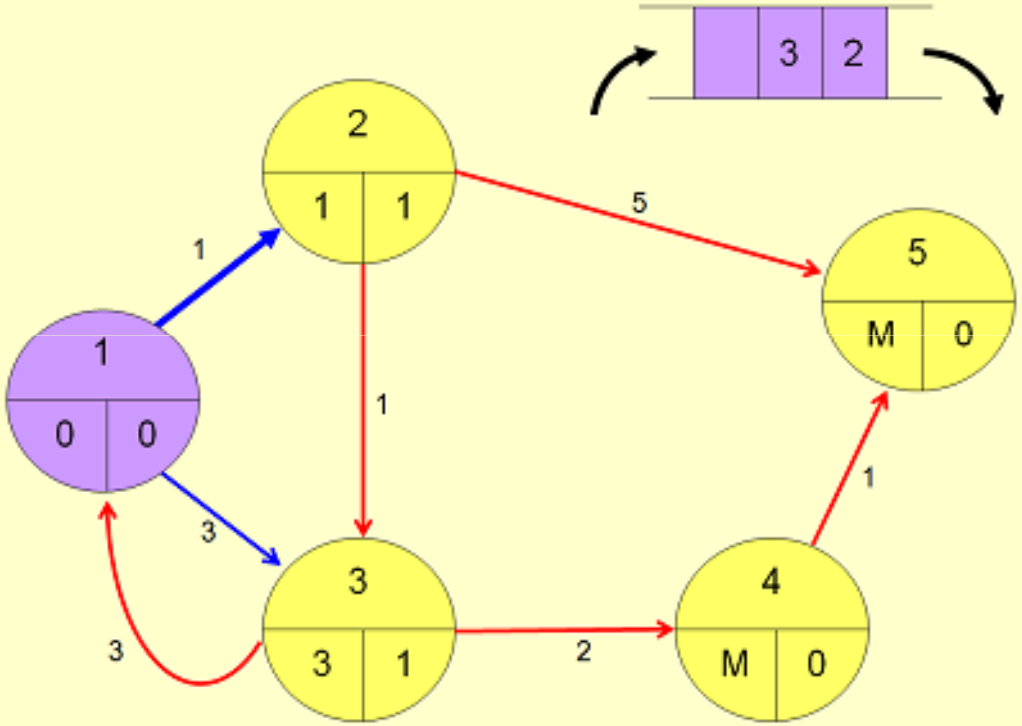

Rute : 1 – 2 – 3 – 4 – 5 dengan beban minimal = 5

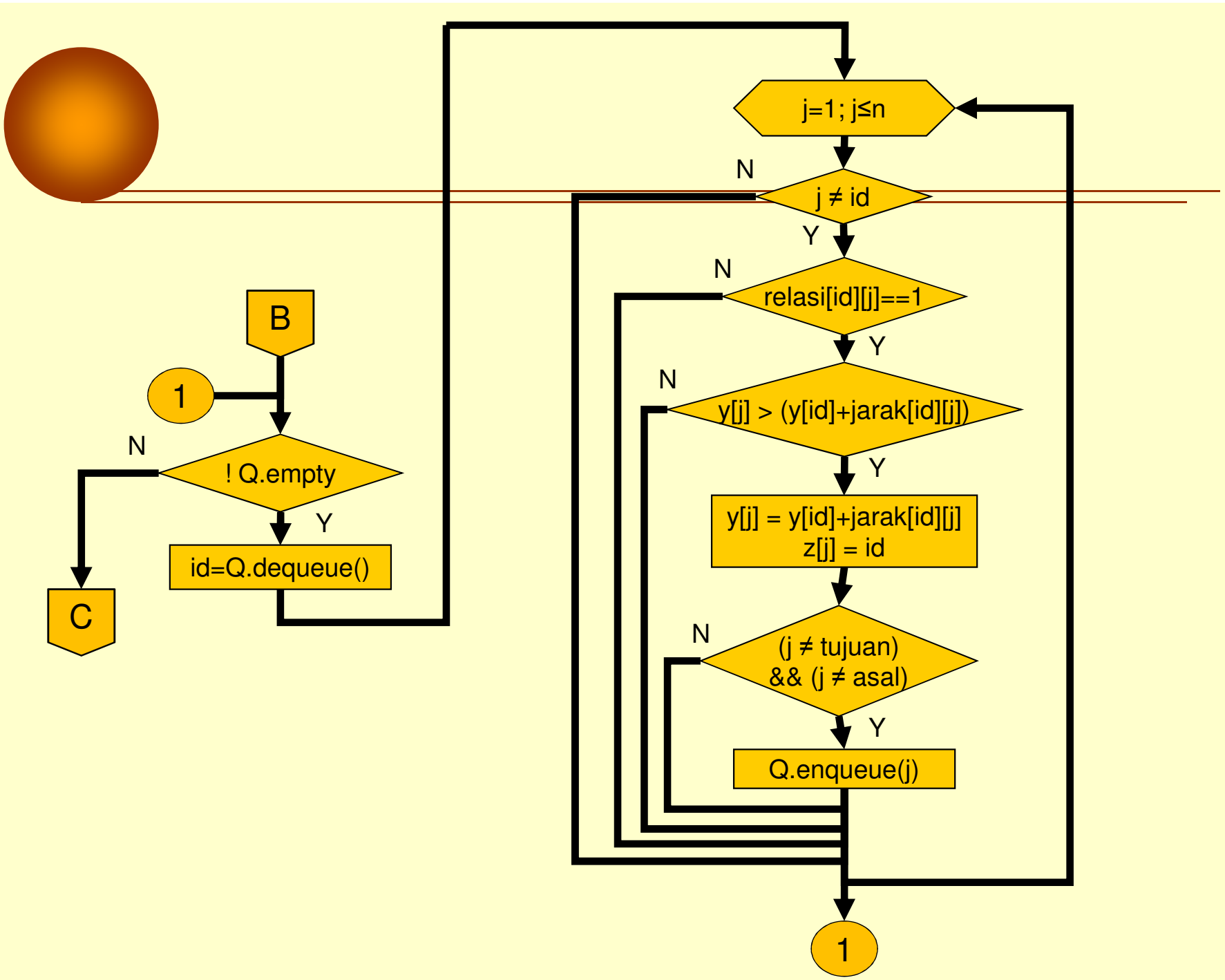

START
n, asal, tujuan,
relasi[][] , jarak[][],
y[], z[], Q, S
Input n, asal, tujuan
relasi[n][n], y[n]
jarak[n][n], z[n]
Input relasi, jarak
y[asal]=0
z[asal]=0

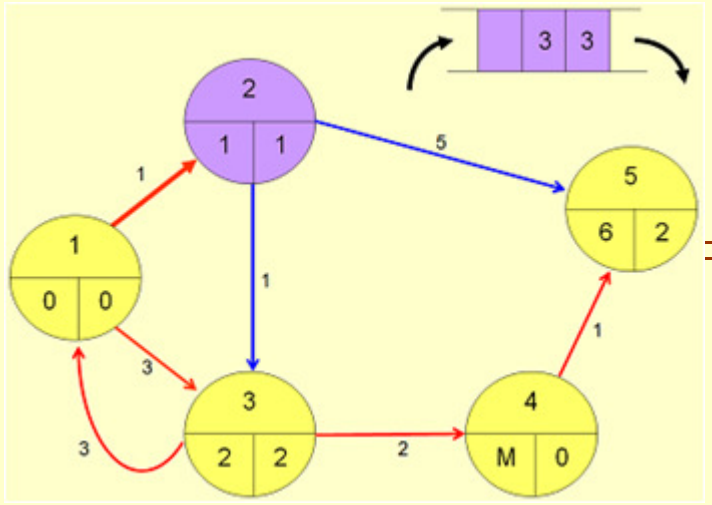

i=1; i ≤n
N
i ≠ asal
Y
y[i]=M
z[i]=0
A

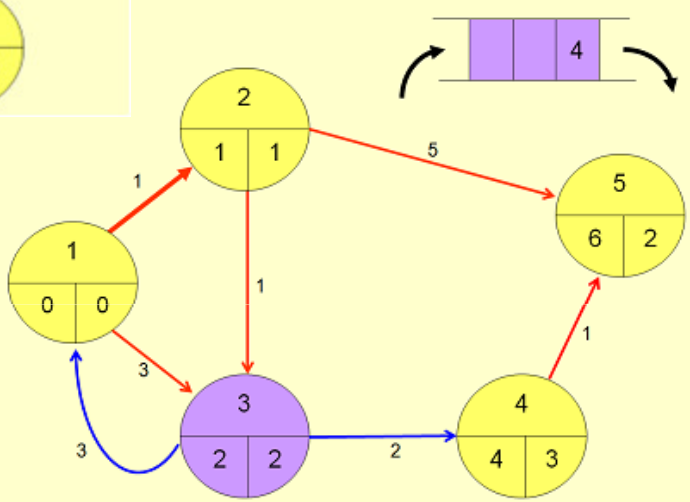

1
0 0
2
M 0
3
M 0
4
M 0
5
M 0
1
5
1
3
3
1
2

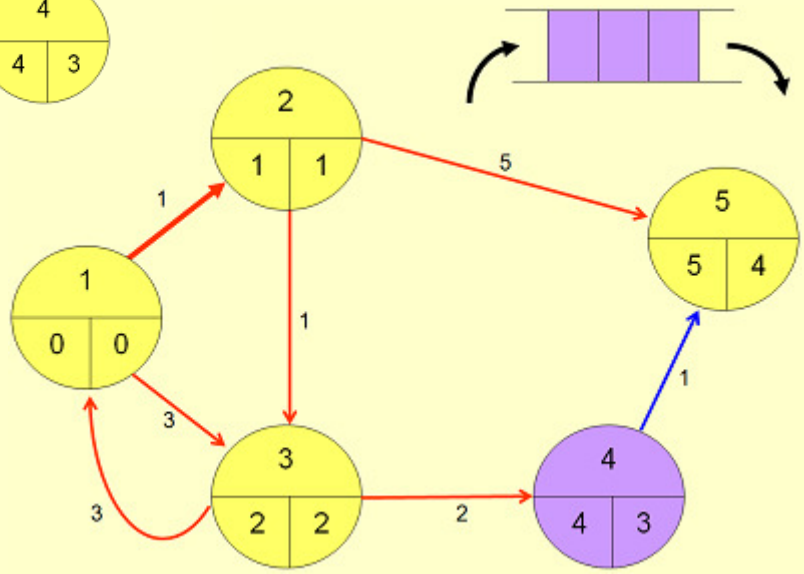

A
j=1; j≤n
N
j ≠ asal
Y
N
relasi[asal][j]==1
Y
y[j] = jarak[asal][j]
z[j] = asal
N
j ≠ tujuan
Y
Q.enqueue(j)
B

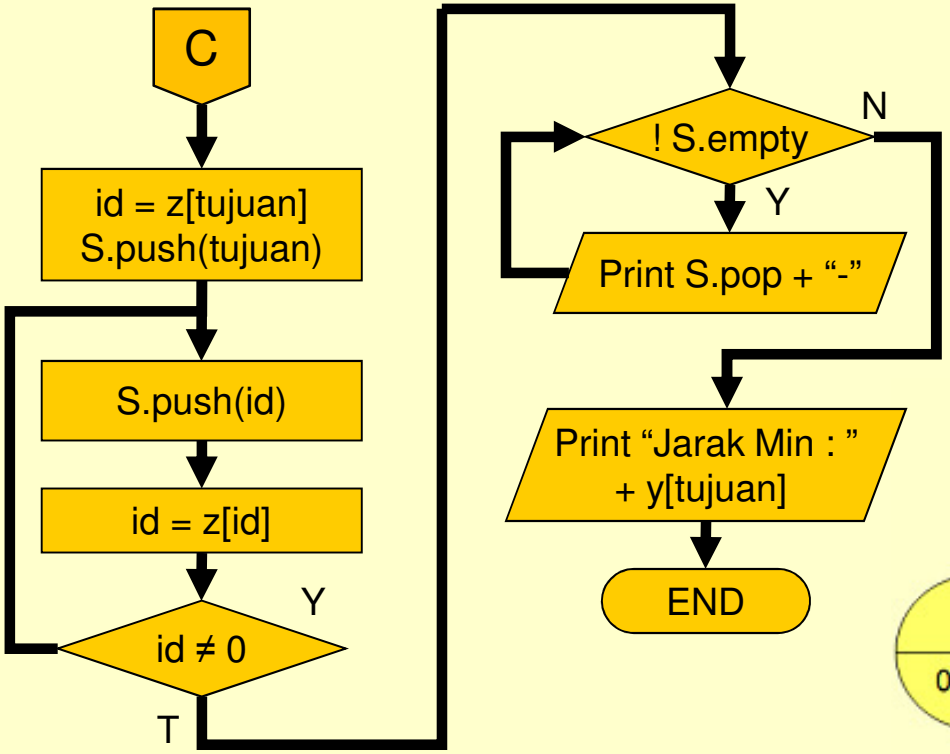

3 2
2
1 1
1
0 0
5
M 0
3
3 1
4
M 0
1
5
1
3
3
2
1

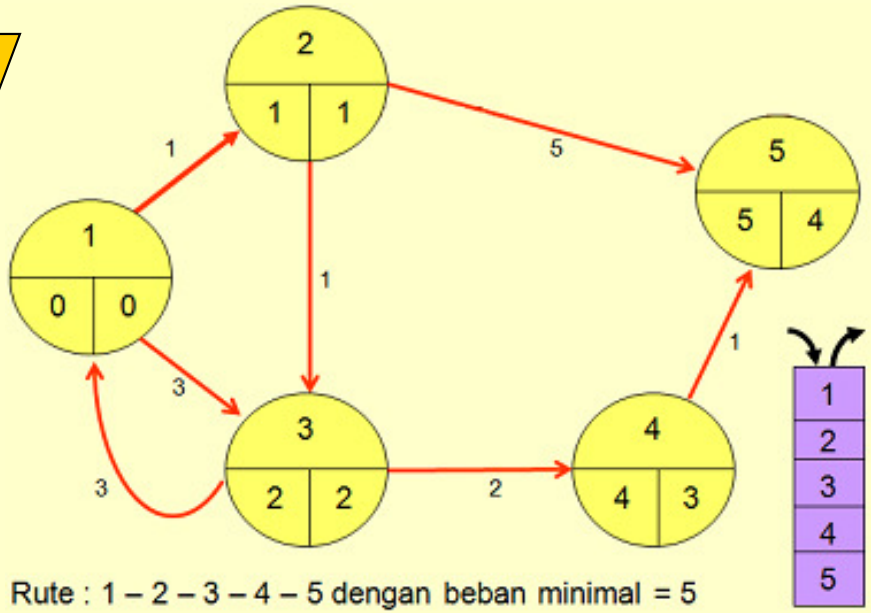

B
1
N
! Q.empty
Y
id=Q.dequeue()
C
j=1; j≤n
N
j ≠ id
Y
N
relasi[id][j]==1
Y
N
y[j] > (y[id]+jarak[id][j])
Y
y[j] = y[id]+jarak[id][j]
z[j] = id
N
(j ≠ tujuan)
&& (j ≠ asal)
Y
Q.enqueue(j)
1


3 3
2
1 1
5
6 2
1
0 0
1
1
3
3
4
M 0
1
3
2 2
2


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


2
1 1
5
5 4
1
0 0
1
1
3
3
4
4 3
1
3
2 2
2




Rute : 1 – 2 – 3 – 4 – 5 dengan beban minimal = 5