$i = [1;2;3;4]$ 1st iteration

no edge $(0;1)$

valid	k	j	current_pathweight	dfs/ drum	drum_final	min_path maxsize
0	0	1	0	[0]	[]	

$i \Rightarrow$ invalid

$i = [2;3;1;4]$ 3th iteration

valid	k	j	curr_pthweight	dfs/ drum	drum_final	min_pth
1	0	2	4	[0;2]	[]	maxsize
1	2	3	7	[0;2;3]	[]	maxsize
1	3	1	17	[0;2;3;1]	[]	maxsize
0	1	4	17	[]	[]	maxsize

$i \Rightarrow$ invalid

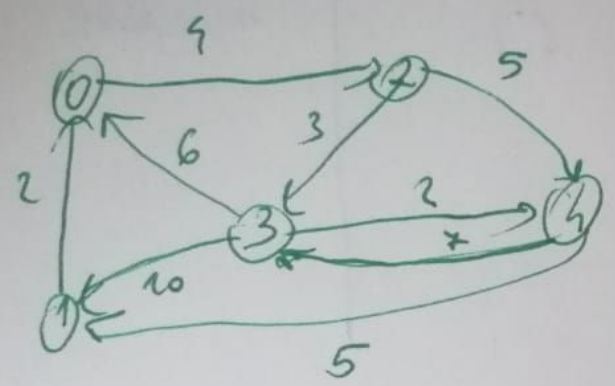
$i = [2;3;4;1]$ 10th iteration

valid	k	j	curr_pthweight	dfs/ drum	drum_final	min_pth
1	0	2	4	[0;2]	[]	maxsize
1	2	3	7	[0;2;3]	[]	maxsize
1	3	4	9	[0;2;3;4]	[]	maxsize
1	4	1	13	[0;2;3;4;1]	[]	maxsize
1	1	0	21	[0;2;3;4;1]	[0;2;3;4;1]	21

$i \Rightarrow$ is valid

There are 24 iterations where we checked all the permutations one by one.

Manual execution:



1) Adjoney matrix is created

$$\begin{pmatrix} 0 & 0 & 4 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 5 \\ 6 & \infty & 0 & 0 & 7 \\ 0 & 5 & 0 & 2 & 0 \end{pmatrix}$$

solution: 0-2-3-4-1-0
cost: 21

2) initialization

~~vrt = empty list~~

drum = [0]

vrt: [1, 2, 3, 4]

min path = ~~max_value~~ maxsize

next_permutation = all the permutations of vrt list.

3) parsing through all the permutations (there are 24 permutations) and the solution is the 10th permutation