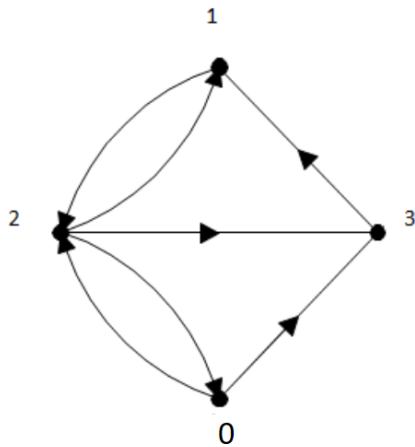


TUGAS PRAKTIKUM STRUKDAT PERTEMUAN 11

Jumat, 12 Mei 2023

1. Notasikan graph berikut dalam bentuk Adjacency list, edge list, dan adjacency matriks.

a.



Jawab :

- Adjacency list

0	2	3	
1	2	3	
2	0	1	3
3	0	1	2

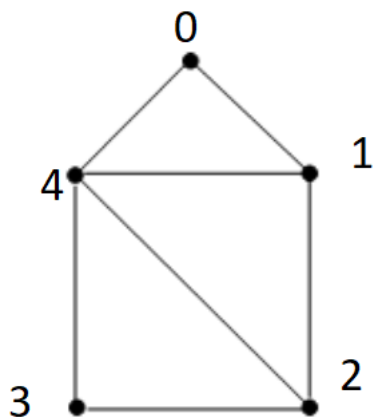
- Edge list

[(0,3),(0,2),(2,3),(2,0),(2,1),(1,2),(3,1)]

- Adjacency matrix

	0	1	2	3
0	0	0	1	1
1	0	0	1	1
2	1	1	0	1
3	1	1	0	0

b.



Jawab :

- Adjacency list

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

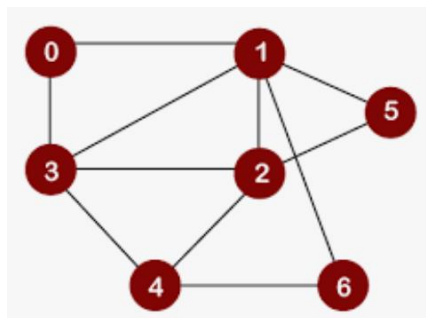
- Edge list

[(0,1),(0,4),(1,2),(1,4),(2,1),(2,3),(2,4),(3,2),(3,4),(4,0),(4,1),(4,2),(4,3)]

- Adjacency matrix

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

2. Diberikan graph berikut, telusurilah setiap verteknya menggunakan algoritma: (a) DFS dan (b) BFS (starting vertexnya vertex 0). Sebagai catatan, jika pada satu titik ada dua opsi yang setara maka dahulukanlah penelusuran vertex dengan no ID yang lebih kecil.



Jawab :

DFS

- Code

```
dfs.py > ...
1 #Wildan Holik
2 #J0403221025
3
4 graph = {
5     '0' : set (['3','1']),
6     '1' : set (['0','5','2','3']),
7     '2' : set (['3','1','5','4']),
8     '3' : set (['0','2','4','1']),
9     '4' : set (['3','2','6']),
10    '5' : set (['2','1']),
11    '6' : set (['4','1'])
12 }
13
14 checkpoint = set()
15
16 def dfs(checkpoint,graph,start):
17     if start not in checkpoint:
18         print(start)
19         checkpoint.add(start)
20         for neighbour in graph[start]:
21             dfs(checkpoint, graph, neighbour)
22
23 dfs([checkpoint, graph, '0'])
24
```

- Output

```
dfs.py > ...
1 #Wildan Holik
2 #J0403221025
3
4 graph = {
5     '0' : set (['3','1']),
6     '1' : set (['0','5','2','3']),
7     '2' : set (['3','1','5','4']),
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS D:\Algoritma dan Struktur Data\Pert11> C:\Users\Asus\AppData\Local\Programs\Python\Python310\python.exe "d:/Algoritma dan Struktur Data/Pert11/dfs.py"

0
1
2
4
6
3
5
PS D:\Algoritma dan Struktur Data\Pert11>

Jawab :

BFS

- Code

```
1 #Wildan Holik
2 #J0403221025
3
4 graph = {
5     '0' : set (('3','1')),
6     '1' : set (('0','5','2','3')),
7     '2' : set (('3','1','5','4')),
8     '3' : set (('0','2','4','1')),
9     '4' : set (('3','2','6')),
10    '5' : set (('2','1')),
11    '6' : set (('4','1'))
12 }
13
14 def bfs_lintasan_terpendek(graph, mulai,tujuan):
15     explored=[]
16     queue=[mulai]
17
18     if mulai == tujuan :
19         return "Awal adalah tujuan"
20     while queue :
21         jalur = queue.pop(0)
22         node = jalur[-1]
23
24         if node not in explored :
25             neighbour = graph[node]
26
27             for neighbour in neighbour :
28                 jalur_baru = list(jalur)
29                 jalur_baru.append(neighbour)
30                 queue.append(jalur_baru)
31
32                 if neighbour == tujuan :
33                     return jalur_baru
34
35             explored.append(node)
36     return "Mohon maaf node yang dicari tidak ada"
37
38 mulai = input("masukan awal : ")
39 tujuan = input("masukan akhir : ")
40 print(bfs_lintasan_terpendek(graph, mulai,tujuan))
```

- Output

```
1 #Wildan Holik
2 #J0403221025
3
4 graph = {
5     '0' : set (('3','1')),
6     '1' : set (('0','5','2','3')),
7     '2' : set (('3','1','5','4')),
8     '3' : set (('0','2','4','1')),
9     '4' : set (('3','2','6')),
10    '5' : set (('2','1')),
11    '6' : set (('4','1'))
12 }
```

PS D:\Algoritma dan Struktur Data\Port11 & C:\Users\Aous\AppData\Local\Programs\Python\Python110\python.exe "d:/Algoritma dan Struktur Data/Port11/bfs.py"

masukan awal : 0

masukan akhir : 5

['0', '1', '5']

PS D:\Algoritma dan Struktur Data\Port11: