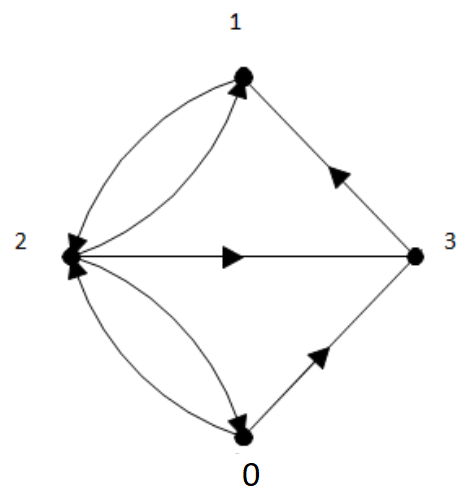
**TUGAS PRAKTIKUM STRUKDAT PERTEMUAN 11**

Jumat, 12 Mei 2023

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



Jawab:

* Adjacenty list

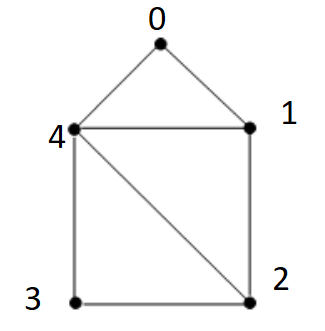
|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 2 | 3 |  |
| 1 | 2 | 3 |  |
| 2 | 0 | 1 | 3 |
| 3 | 0 | 1 | 2 |

* 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 |



Jawab:

* Adjacency list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 4 |  |  |
| 1 | 0 | 2 | 4 |  |
| 2 | 1 | 3 | 4 |  |
| 3 | 2 | 4 |  |  |
| 4 | 0 | 1 | 2 | 3 |

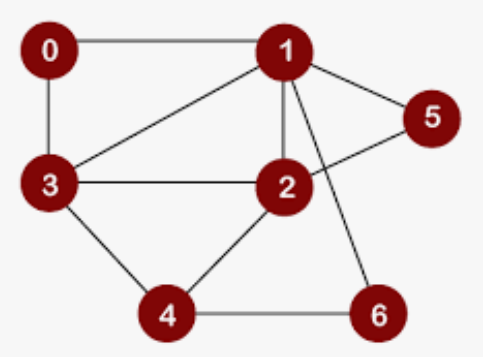
* Edge list

[ (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 list

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 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 |

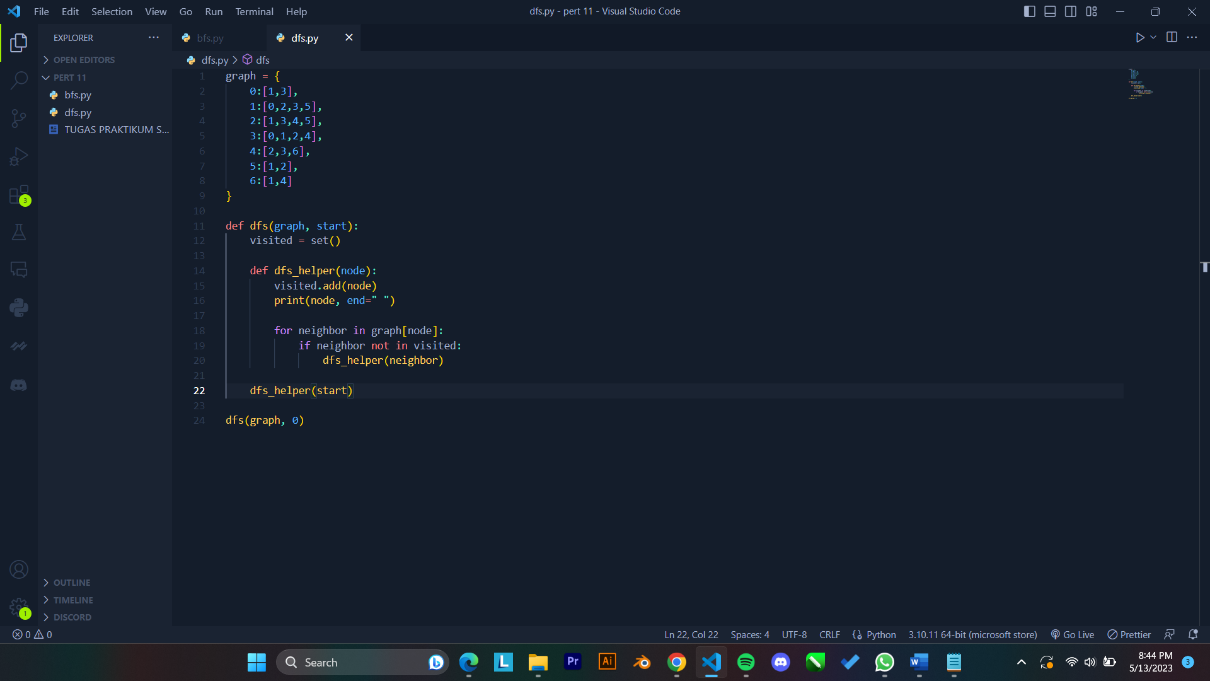
1. Diberikan graph berikut, telusurilah setiap verteknya menggunakan algoritma: (a) DFS dan (b) BFS (starting vertexnya vertex 0). Sebagai catatan, jika pada satu titik ada dua opsi yang setara maka dahulukanlah penelusuran vertex dengan no ID yang lebih kecil.



Jawab :

* DFS

1. Mulai dari node 0, tandai node 0 sebagai visited (dikunjungi).
2. Pilih node tetangga dengan ID terkecil dari node 0 yang belum dikunjungi, yaitu node 1.
3. Kunjungi node 1 dan tandai sebagai visited.
4. Pilih node tetangga dengan ID terkecil dari node 1 yang belum dikunjungi, yaitu node 3.
5. Kunjungi node 3 dan tandai sebagai visited.
6. Tidak ada node tetangga dari node 3 yang belum dikunjungi, kembali ke node 1.
7. Pilih node tetangga dengan ID terkecil dari node 1 yang belum dikunjungi, yaitu node 4.
8. Kunjungi node 4 dan tandai sebagai visited.
9. Tidak ada node tetangga dari node 4 yang belum dikunjungi, kembali ke node 1.
10. Pilih node tetangga dengan ID terkecil dari node 1 yang belum dikunjungi, yaitu node 2.
11. Kunjungi node 2 dan tandai sebagai visited.
12. Pilih node tetangga dengan ID terkecil dari node 2 yang belum dikunjungi, yaitu node 6.
13. Kunjungi node 6 dan tandai sebagai visited.
14. Tidak ada node tetangga dari node 6 yang belum dikunjungi, kembali ke node 2.
15. Pilih node tetangga dengan ID terkecil dari node 2 yang belum dikunjungi, yaitu node 5.
16. Kunjungi node 5 dan tandai sebagai visited.
17. Tidak ada node tetangga dari node 5 yang belum dikunjungi, kembali ke node 2.
18. Tidak ada node tetangga dari node 2 yang belum dikunjungi, kembali ke node 0.
19. Selesai.

Hasilnya (0,1,2,3,4,6,5)

* BFS

1. Tentukan starting vertex dan target vertex (jika ingin mencari jalur dari satu vertex ke vertex lain).
2. Tandai starting vertex sebagai dikunjungi.
3. Masukkan starting vertex ke dalam queue.
4. Loop sampai queue kosong:

a. Ambil node dari depan queue.

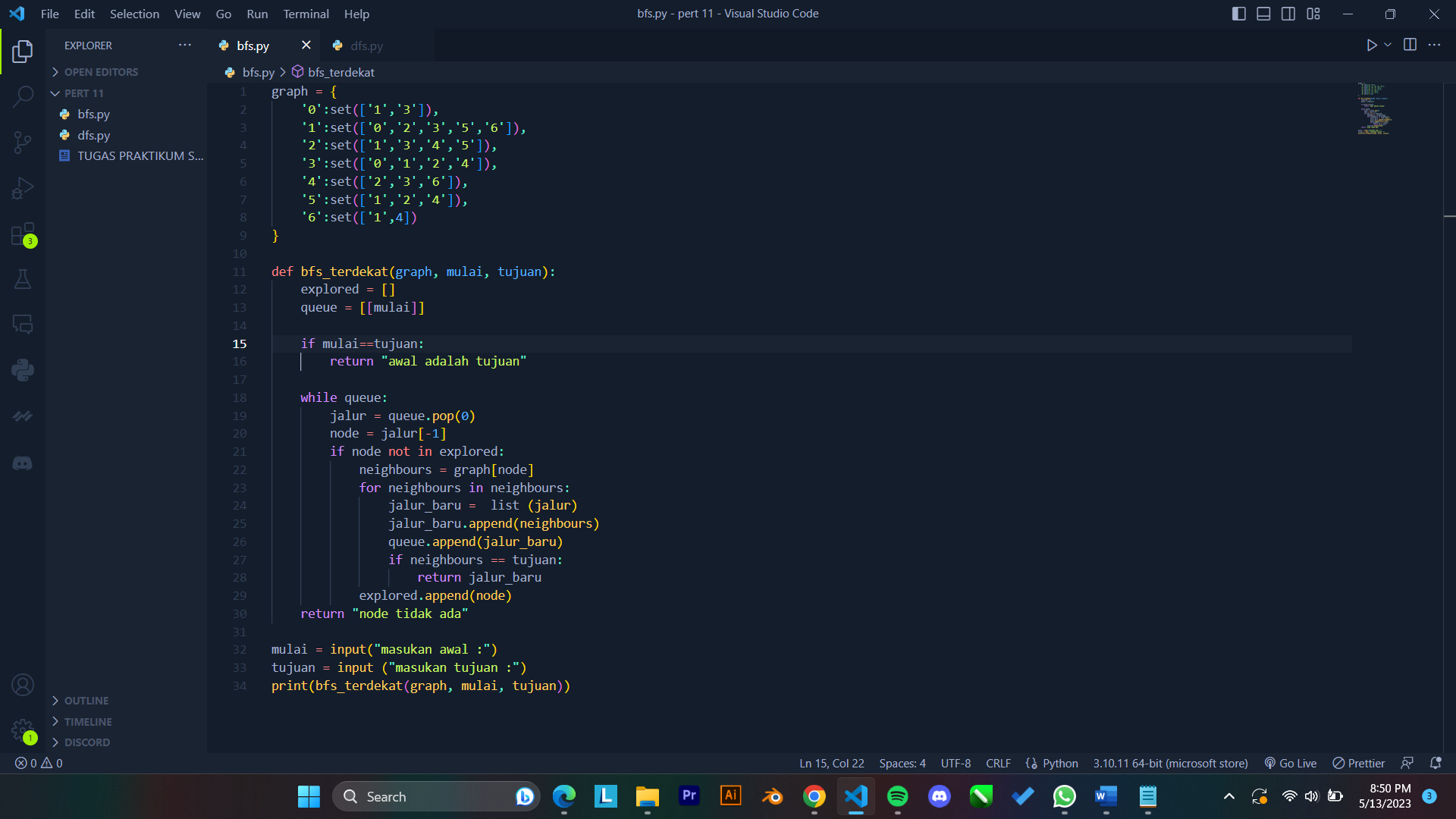
b. Cek apakah node sama dengan target vertex (jika ingin mencari jalur ke target). i. Jika ya, selesai.

c. Tandai node sebagai dikunjungi.

d. Tambahkan tetangga node yang belum dikunjungi ke dalam queue.

1. Jika queue kosong dan target vertex belum ditemukan, berarti target tidak dapat dicapai dari starting vertex.

Code :



Output :

