|  |  |
| --- | --- |
| START  IMPORT necessary libraries  DEFINE class Tugas  METHOD main  CREATE Scanner object 'scanner'  CREATE List 'hasil' to store results  WHILE true  READ 'jumlahKota' from input  READ 'jumlahAliansi' from input    IF 'jumlahKota' == 0 AND 'jumlahAliansi' == 0  BREAK out of the loop  CREATE List 'daftarAliansi' to store alliances    FOR i = 0 to 'jumlahAliansi' - 1  READ 'kotaDalamAliansi' from input  CREATE List 'aliansi'    FOR j = 0 to 'kotaDalamAliansi' - 1  READ city and ADD to 'aliansi'    ADD 'aliansi' to 'daftarAliansi'    CALL 'cariKotaAwal' with 'jumlahKota', 'jumlahAliansi', 'daftarAliansi' and ADD result to 'hasil'    FOR each 'kota' in 'hasil'  PRINT 'kota'    CLOSE 'scanner'    METHOD 'cariKotaAwal' (int n, int c, List 'daftarAliansi') RETURNS int  CREATE Map 'petaAliansiKota'  CREATE Map 'graf'    FOR each 'indeksAliansi' in 'daftarAliansi'  GET 'aliansi' from 'daftarAliansi' at 'indeksAliansi'    FOR i = 0 to size of 'aliansi' - 1  FOR j = i + 1 to size of 'aliansi' - 1  CREATE keys 'kunci1' and 'kunci2'  ADD connections to 'graf' for both 'kunci1' and 'kunci2'    FOR each 'indeksAliansi' in 'daftarAliansi'  GET 'aliansi' from 'daftarAliansi' at 'indeksAliansi'    FOR each 'kota' in 'aliansi'  ADD 'indeksAliansi' to 'petaAliansiKota' for 'kota'    FOR 'kota' = 0 to n - 1  FOR each 'aliansi' in 'petaAliansiKota' for 'kota'  IF 'dapatKunjungiSemuaKota' with 'kota', 'aliansi', n, 'petaAliansiKota', 'graf'  RETURN 'kota'    RETURN -1  METHOD 'dapatKunjungiSemuaKota' (int kotaMulai, int aliansiMulai, int n, Map 'petaAliansiKota', Map 'graf') RETURNS boolean  CREATE Queue 'antrian'  ADD new Node with 'kotaMulai', 'aliansiMulai', and initial visited set and path to 'antrian'    WHILE 'antrian' is not empty  POLL 'node' from 'antrian'  GET 'kotaSekarang', 'aliansiSekarang', 'sudahDikunjungi', 'jalur' from 'node'    IF size of 'sudahDikunjungi' == n  RETURN true    GET 'aliansiSelanjutnya' from 'petaAliansiKota' for 'kotaSekarang'    FOR each 'aliansi' in 'aliansiSelanjutnya'  IF 'aliansi' is not 'aliansiSekarang' OR size of 'aliansiSelanjutnya' == 1  CREATE 'kunci'  GET 'tetangga' from 'graf' for 'kunci'    FOR each 'tetanggaStr' in 'tetangga'  PARSE 'kotaTetangga' and 'aliansiTetangga' from 'tetanggaStr'    IF 'kotaTetangga' not in 'sudahDikunjungi'  CREATE new visited set 'dikunjungiBaru' and add 'kotaTetangga'  CREATE new path 'jalurBaru' and add 'kotaTetangga'  ADD new Node with 'kotaTetangga', 'aliansiTetangga', 'dikunjungiBaru', 'jalurBaru' to 'antrian'    RETURN false  DEFINE class Node  CONSTRUCTOR Node (int kota, int aliansi, Set 'dikunjungi', List 'jalur')  ASSIGN 'kota', 'aliansi', 'dikunjungi', 'jalur'    END  **Sehingga notasi Big O pada kode ini adalah O(n3)** | O(1)  O(1)  O(1)  O(1)  O(1)  O(1)  O(1)  O(n)  O(1)  O(1)  O(n-1)  O(1)  O(1)  O(n-1)  O(1)  O(1)  O(n)  O(n)  O(1)  O(1)  O(1)  O(1)  O(n)  O(1)  O(n-1)  O(n-1)  O(1)  O(1)  O(n)  O(1)  O(n)  O(1)  O(n)  O(n)  O(1)  O(1)  O(1)  O(1)  O(1)  O(n)  O(1)  O(n3)  O(1)  O(1)  O(1)  O(n)  O(1)  O(1)  O(1)  O(n)  O(1)  O(1)  O(1)  O(1)  O(1)  O(1)  O(1)  O(1) |