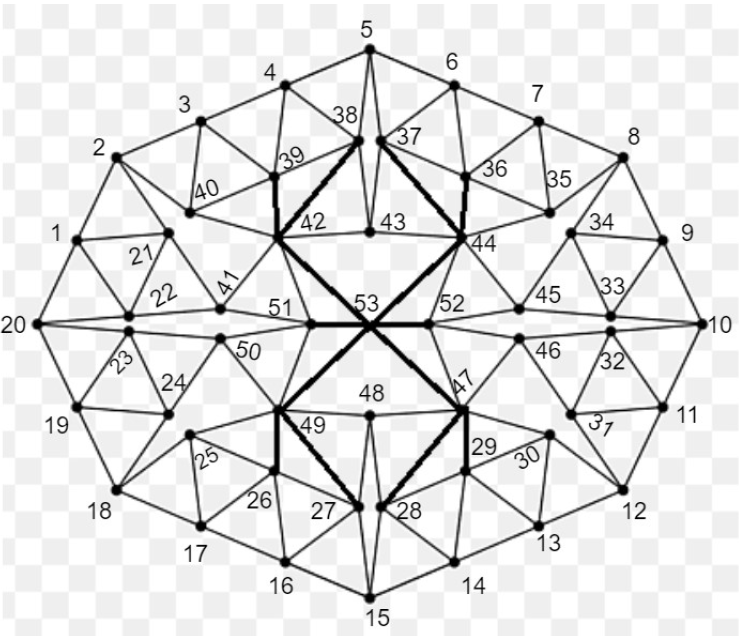
For any set of vertices U, let G[U] denote the subgraph of G induced by U. ​

* The core of G, denoted core(G), sometimes also called the k-core of G, is the subgraph induced by a subset of vertices of G whose ? is maximum.​ (2pts)

Answer: peel value

* The core(G) can be computed by iteratively removing vertices of minimum degree and updating the neighbor's degree accordingly. The core of the graph attached below is {…?...}​ (5pts)

Answer: The core of the graph is 4-core of G which is the full graph G.



* A graph Fk is a fixed point of degree peeling k if core(Fk) ? V(Fk) and the ? of each vertex in Fk is k. ​(3pts)

Answer: 1) is equal to and 2) peel value

* Can you find a fixed point with peel value k = 2 in the graph G attached below?​ (5pts). If so, draw the subgraph that is a fixed point of peel value 2. If there is no such subgraph explain why.

Answer: After removing vertices with degree = 4, we got peel value = 2.

Shape

Description automatically generated

Find a Wave with seed set S0 = {1, 2, 3, 4, 5} in the graph G attached below.​ (10pts) and Property P being degree being less than or equal to 4.

|  |  |
| --- | --- |
| S0 = {1, 2, 3, 4, 5}  frag(S0) = {(1, 2), (1, 20), (1, 21), (1, 22),  (2, 3), (2, 21), (2, 40),  (3, 4), (3, 39), (3, 40),  (4, 5), (4, 38), (4, 39),  (5, 6), (5, 37), (5, 38)}  S1 = BP(S0)= S0 + {6, 20, 21, 22, 37, 38, 39, 40}  S1 = {1, 2, 3, 4, 5, 6, 20, 21, 22, 37, 38, 39, 40}  frag(S1) = frag(S0) +  {(6, 7), (6, 36), (6, 37),  (20, 19), (20, 22), (20, 23),  (21, 22), (21, 41),  (22, 41),  (37, 36), (37, 43), ~~(37, 44),~~  (38, 39), (38, 42), ~~(38, 43),~~  (39, 40), ~~(39, 42),~~  (40, 42)}  S2 = BP(S1)= S1 + {7, 19, 23, 36, 41, 42, 43}  S2 = {1, 2, 3, 4, 5, 6, 7, 19, 20, 21, 22, 23, 36, 37, 38, 39, 40, 41, 42, 43}  frag(S2) = frag(S1) +  {(7, 8), (7, 35), (7, 36),  (19, 18), (19, 23), (19, 24),  (23, 24), (23, 50),  (36, 35), ~~(36, 37), (36, 44),~~  (41, 42), (41, 51),  (42, 43), ~~(42, 51), (42, 53),~~  (43, 44)}  S3 = BP(S2)= S2 + {8, 18, 24, 35, 44, 50, 51}  S3 = {1, 2, 3, 4, 5, 6, 7, 8, 18, 19, 20, 21, 22, 23, 24, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 50, 51}  Note: 1. Highlight is a new node  2. Cross out edge is an edge that makes node not satisfy property P | Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 1), (20, 1), (21, 2), (22, 1), (37, 1), (38, 2), (39, 2), (40, 2)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 1), (19, 1), (20, 4), (21, 4), (22, 4), (23, 1), (36, 2), (37, 4), (38, 4), (39, 4), (40, 4), (41, 2), (42, 2), (43, 1)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 1), (18, 1), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 2), (35, 2), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 1), (50, 1), (51, 1)] |

|  |  |
| --- | --- |
| frag(S3) = frag(S2) +  {(8, 9), (8, 34), (8, 35),  (18, 17), (18, 24), (18, 25),  (24, 50),  (35, 44),  (44, 45), (44, 52), ~~(44, 53),~~  (50, 49), (50, 51),  (51, 49), (51, 53)}  S4 = BP(S3)= S3 + {9, 17, 25, 34, 45, 49, 52, 53}  S4 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 17, 18, 19, 20, 21, 22, 23, 24, 25, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 49, 50, 51, 52, 53}  frag(S4) = frag(S3) +  {(9, 10), (9, 33), (9, 34),  (17, 16), (17, 25), (17, 26),  (25, 26), (25, 49),  (34, 33), (34, 45),  (45, 33), (45, 52),  (49, 26), ~~(49, 27), (49, 48), (49, 53),~~  (52, 46), (52, 47), ~~(52, 53)~~  (53, 47)}  S5 = BP(S4)= S1 + {10, 16, 26, 33, 46, 47}  S5 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 49, 50, 51, 52, 53}  frag(S5) = frag(S4) +  {(10, 11), (10, 32), (10, 33),  (16, 15), (16, 26), (16, 27),  ~~(26, 27),~~  (46, 31), (46, 32), (46, 47),  (47, 28), ~~(47, 29), (47, 30), (47, 48)~~}  S6 = BP(S5)= S5 + {11, 15, 27, 28, 31, 32}  S6 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 49, 50, 51, 52, 53} | Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 1), (17, 1), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 1), (34, 1), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 1), (49, 2), (50, 4), (51, 4), (52, 1), (53, 1)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 1), (16, 1), (17, 4), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 4), (26, 3), (33, 3), (34, 4), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 4), (46, 1), (47, 2), (49, 4), (50, 4), (51, 4), (52, 4), (53, 2)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 1), (15, 1), (16, 4), (17, 4), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 4), (26, 4), (27, 1), (28, 1), (31, 1), (32, 2), (33, 4), (34, 4), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 4), (46, 4), (47, 4), (49, 4), (50, 4), (51, 4), (52, 4), (53, 2)] |

|  |  |
| --- | --- |
| frag(S6) = frag(S5) +  {(11, 12), (11, 31), (11, 32),  (15, 14), (15, 27), (15, 28),  (27, 48),  (28, 14), (28, 29), ~~(28, 48),~~  (31, 12), (31, 32)}  S7 = BP(S6)= S6 + {12, 14, 29, 48}  S7 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53}  frag(S7) = frag(S6) +  {(12, 13), (12, 30),  (14, 13), (14, 29),  (29, 13), (29, 30)}  S8 = BP(S7)= S7 + {13, 30}  S8 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53}  frag(S8) = frag(S7) +  {(13, 30)}  S9 = BP(S8)= S8 + {}  S9 = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53} | Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 4), (12, 2), (14, 2), (15, 4), (16, 4), (17, 4), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 4), (26, 4), (27, 3), (28, 4), (29, 1), (31, 4), (32, 4), (33, 4), (34, 4), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 4), (46, 4), (47, 4), (48, 1), (49, 4), (50, 4), (51, 4), (52, 4), (53, 2)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 4), (12, 4), (13, 3), (14, 4), (15, 4), (16, 4), (17, 4), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 4), (26, 4), (27, 3), (28, 4), (29, 4), (30, 2), (31, 4), (32, 4), (33, 4), (34, 4), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 4), (46, 4), (47, 4), (48, 1), (49, 4), (50, 4), (51, 4), (52, 4), (53, 2)]  Degree: [(1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 4), (12, 4), (13, 4), (14, 4), (15, 4), (16, 4), (17, 4), (18, 4), (19, 4), (20, 4), (21, 4), (22, 4), (23, 4), (24, 4), (25, 4), (26, 4), (27, 3), (28, 4), (29, 4), (30, 3), (31, 4), (32, 4), (33, 4), (34, 4), (35, 4), (36, 4), (37, 4), (38, 4), (39, 4), (40, 4), (41, 4), (42, 4), (43, 3), (44, 4), (45, 4), (46, 4), (47, 4), (48, 1), (49, 4), (50, 4), (51, 4), (52, 4), (53, 2)] |

Chart

Description automatically generated

Red edges are the edges in a wave.

* The iterative decomposition of the edges of a graph G into edge maximal fixed points is computed as follows,​
* Step 0 : Current\_graph <--- G.​
* Step 1: Compute the set of vertices in Core(Current\_graph)​
* Step 2: Take the edges in the subgraph induced by Core(Current\_graph and label these edges with the peel value of Core(Current\_graph).​
* Step 3: Remove the edges in the subgraph induced by Core(Current\_graph) and repeat from step 1 with Current\_graph <--- Edges(Current\_graph) – Edges(Core(Current\_graph)).​
* Question: For graph G attached below find the iterative decomposition of the edges of G using the algorithm given above.​ (15pts)
* Question: Draw manually the graph cities for the iterative edge decomposition fund in the previous question. (10pts)