FORBIDDEN SUBGRAPH (Test data description)

Author: Francisco Zaragoza (Mexico)

TEST CASES 1 AND 2: In these two test cases *H* is the disjoint union of two edges. This are easy cases because in general, the optimal answer can be obtained by finding a vertex of maximum degree. However, there is an exception to this: if the maximum degree is 2 and *G* contains a triangle then the best thing to do is to take this triangle as the output. In test case 1, *G* has 100 vertices and is the disjoint union of some triangles and edges. The greedy strategy gets only 1 edge, maximum degree is 2, and optimal is 3 edges. In test case 2, *G* has 87 vertices, 406 edges and maximum degree 82 (greedy gets 6).

TEST CASES 3 AND 4: In these two test cases *H* is a path with two edges. Therefore any feasible solution is a matching, and the optimal solution must be a maximum cardinality matching. In test case 3, *G* has 138 vertices and 493 edges. Greedy gets 4 and there are matchings with 52 edges. In test case 4, *G* has 496 vertices and 11654 edges. Greedy gets 1 edge and there are matchings with 134 edges.

TEST CASES 5 AND 6: In these two test cases *H* is a star with three spokes (three edges with a common vertex.) Therefore any feasible solution is the disjoint union of some cycles and paths. In test case 5, *G* has 23 vertices and 71 edges. Greedy gets 2 and there are solutions (obtained with a modified matching algorithm) that have 23 edges. In test case 6, *G* has 211 vertices and 4100 edges. Greedy gets 2 and there are solutions with 126 edges.

TEST CASES 7 AND 8: In these two test cases *H* is a triangle. Maximum size triangle free subgraphs is a hard problem and heuristics would be good to use here. In test case 7, *G* is a double-wheel, that is, a cycle and two additional vertices connected to every vertex of the cycle and among themselves. The greedy strategy chooses the edge joining these two vertices and limits severely the number of other edges that can be chosen. A more careful analysis avoids choosing this edge and obtains a much better solution. In test case 8, *G* has 74 vertices and 301 edges. Greedy gets 31 edges and there are solutions with 195 edges.

TEST CASES 9 AND 10: In these two test cases *H* is a paw, that is, a triangle with a hanging edge. This is another hard problem, but since a paw contains a triangle, any code written for the triangle case helps here. In test case 9, *G* has 211 vertices and 3541 edges. Greedy gets 140 and there are solutions with 2868 edges. In test case 10, *G* has 645 vertices and 13979 edges. Greedy gets 541 and there are solutions with 12125 edges.