

QueryId	Pattern	Shape	Type constraints	Label constraints
27	(a)-->(b), (a)-->(c)	*		
28	(a:Person)-->(b), (a)-->(c:Person)	*		yes
29	(a:Message)-->(b), (a)-->(c), (a)-->(d), (a)-->(e:Message)	*		yes
30	(a)-->(b), (a)<--(c:Company)	*		yes
31	(a)-->(b:Post), (a)-->(c:Tag), (a)<--(d:City)	*		yes
32	(a)-->(b:Continent), (a)-->(c:Place), (a)-->(d:Place), (a)-->(e:Place)	*		yes
33	(m)-[:IS_LOCATED_IN]->(l), (m)-[:REPLY_OF]->(m2), (m)-[:HAS_CREATOR]->(p)	*	yes	yes
34	(f)-[:HAS_MODERATOR]->(mod), (f)-[:HAS_MEMBER]->(mem), (f)-[:HAS_TAG]->(x)	*	yes	yes
35	(c1)-[:REPLY_OF]->(c2)-[:STUDY_AT]->(u), (c2)-->(x1), (x2)-->(c2)	*	yes	
36	(p)-[:LIKES]->(m)<-[:REPLY_OF]-(c), (m)-[:HAS_TAG]->(t), (m)-[:IS_LOCATED_IN]->(l)	*	yes	yes
37	(m)-[:IS_LOCATED_IN]->(l:Place), (m:Comment)-[:REPLY_OF]->(m2:Message), (m)-[:HAS_CREATOR]->(p:Person)	*	yes	yes (redundant)
38	(f:Forum)-[:HAS_MODERATOR]->(mod:Person), (f)-[:HAS_MEMBER]->(mem:Person), (f)-[:HAS_TAG]->(x:Tag)	*	yes	yes (redundant)
39	(p:Person)-[:LIKES]->(m:Message)<-[:REPLY_OF]-(c:Comment), (m)-[:HAS_TAG]->(t:Tag), (m)-[:IS_LOCATED_IN]->(l:Country)	*	yes	yes (redundant)
40	(c:Country:Continent)<-[:IS_LOCATED_IN]-(x1), (c)<-[:IS_LOCATED_IN]-(x2), (c)<-[:IS_LOCATED_IN]-(x4), (c)<-[:IS_LOCATED_IN]-(x5)	*	yes	yes (disjoint)