

Especificació dels jocs de prova pel sistema global

Grup 14.2

Índex

1	addNode	2
2	addEdge	3
3	addWrongEdge	4
4	removeNode	5
5	removeEdge	6
6	dupNames	7
7	jpAddRemRem	8
8	jpEdges	9
9	jpInvalidTypes	10
10	jpNeighAfterRemNode	11
11	query1to1	12
12	query1toN	13
13	queryNtoN	14
14	queryByReference	15
15	queryAndResult	16

1 addNode

Descripció

Els casos d'ús que es proven són:

- Afegir un node al graf.

S'afegeix un node.

Objectius

Afegir un node al graf.

Entrada

Al fitxer `addNode.in`.

Sortida

Al fitxer `addNode.out`.

Resultat

Correcte.

2 addEdge

Descripció

Els casos d'ús que es proven són:

- Afegir un node al graf.
- Afegir una aresta al graf.

S'afegeix un node autor i un altre paper i una aresta entre ells.

Objectius

Afegir arestes.

Entrada

Al fitxer `addEdge.in`.

Sortida

Al fitxer `addEdge.out`.

Resultat

Correcte.

3 addWrongEdge

Descripció

Els casos d'ús que es proven són:

- Afegir una aresta al graf (cas de tipus incompatibles).

S'afegeix un node autor i un altre term, i s'intenta afegir una aresta entre ells.

Objectius

Comprovar l'avís d'aresta errònia.

Entrada

Al fitxer `addWrongEdge.in`.

Sortida

Al fitxer `addWrongEdge.out`.

Resultat

Correcte.

4 removeNode

Descripció

Els casos d'ús que es proven són:

- Afegir un node al graf.
- Esborrar un node del graf.

S'afegeix un node i després s'esborra.

Objectius

Esborrar un node del graf.

Entrada

Al fitxer `removeNode.in`.

Sortida

Al fitxer `removeNode.out`.

Resultat

Correcte.

5 removeEdge

Descripció

Els casos d'ús que es proven són:

- Esborrar una aresta.

S'afegeixen dos nodes i una aresta entre ells. Després s'esborra l'aresta.

Objectius

Esborrar arestes.

Entrada

Al fitxer `removeEdge.in`.

Sortida

Al fitxer `removeEdge.out`.

Resultat

Correcte.

6 dupNames

Descripció

Els casos d'ús que es proven són:

- Distingir nodes de nom repetit.

S'afegeixen dos nodes amb el mateix nom. Després s'intenta esborrar un i surt un avís per triar quin dels dos s'ha d'esborrar.

Objectius

Distingir nodes de nom repetit.

Entrada

Al fitxer `dupNames.in`.

Sortida

Al fitxer `dupNames.out`.

Resultat

Correcte.

7 jpAddRemRem

Descripció

Els casos d'ús que es proven són:

- Afegir un node al graf.
- Esborrar un node del graf.
- Esborrar un node del graf (cas en que no existeix).

S'afegeix un autor i s'esborra dues vegades. A la segona s'avisava que no existeix cap node.

Objectius

Afegir i esborrar nodes correctament, avisant a l'usuari quan no existeix el node introduït.

Entrada

Al fitxer `jpAddRemRem.in`.

Sortida

Al fitxer `jpAddRemRem.out`.

Resultat

Correcte.

8 jpEdges

Descripció

Els casos d'ús que es proven són:

- Afegir una aresta.
- Consultar nodes i arestes.
- Esborrar una aresta.
- Esborrar una aresta (cas en que no existeix).

S'afegeixen dos autors i un paper, i una aresta entre cada autor i el paper. Es consulten els nodes i arestes. Després s'esborra l'aresta entre el paper i un dels autors, dues vegades: la primera s'esborra amb èxit, a la segona surt un missatge aclarint que l'aresta no existeix. Es torna a consultar l'estat del graf.

Objectius

Afegir i esborrar arestes correctament, avisant a l'usuari quan no existeix l'aresta introduïda. Consultar nodes i arestes.

Entrada

Al fitxer `jpEdges.in`.

Sortida

Al fitxer `jpEdges.out`.

Resultat

Correcte.

9 jpInvalidTypes

Descripció

Els casos d'ús que es proven són:

- Afegir una aresta (entre tipus incompatibles).
- Afegir una aresta (entre un node que existeix i un altre que no).
- Esborrar un node del graf (cas en que no existeix).

Afegeix tres nodes, un paper, un term i una conf. Intenta afegir una aresta entre la conf i el term i no es pot per tipus incompatibles. Intenta afegir una aresta entre el paper i un autor inexistent, i tampoc es permet. Intenta esborrar un autor inexistent i surt un avís.

Objectius

Comprovar si surten avisos quan s'intenten afegir arestes impossibles.

Entrada

Al fitxer `jpInvalidTypes.in`.

Sortida

Al fitxer `jpInvalidTypes.out`.

Resultat

Correcte.

10 jpNeighAfterRemNode

Descripció

Els casos d'ús que es proven són:

- Esborrar un node (que té arestes).
- Consulta de veïns (després d'esborrar algun veï).

Afegeix un node terme i un altre paper, i una aresta entre ells. Despres esborra el node terme i consulta els veïns del paper. El node terme ja no hi apareix.

Objectius

Comprovar si quan s'esborren nodes, s'esborra correctament tot el seu rastre del graf.

Entrada

Al fitxer `jpNeighAfterRemNode.in`.

Sortida

Al fitxer `jpNeighAfterRemNode.out`.

Resultat

Correcte.

11 query1to1

Descripció

Els casos d'ús que es proven són:

- Consulta 1 a 1 (quan el HeteSim és positiu).
- Consulta 1 a 1 (quan el HeteSim és zero).
- Consulta 1 a 1 (quan un dels nodes no existeix).

S'afegeix el graf d'exemple del article de Shi et al. i es prova la consulta 1 a 1 en els casos especificats.

Objectius

Comprovar la funcionalitat de consulta 1 a 1.

Entrada

Al fitxer `query1to1.in`.

Sortida

Al fitxer `query1to1.out`.

Resultat

Correcte.

12 query1toN

Descripció

Els casos d'ús que es proven són:

- Consulta 1 a N.

S'afegeix el graf d'exemple del article de Shi et al. i es prova la consulta 1 a N.

Objectius

Comprovar la funcionalitat de consulta 1 a N.

Entrada

Al fitxer `query1toN.in`.

Sortida

Al fitxer `query1toN.out`.

Resultat

Correcte.

13 queryNtoN

Descripció

Els casos d'ús que es proven són:

- Consulta N a N (quan hi ha tipus intermig).
- Consulta N a N (quan no hi ha tipus intermig).

S'afegeix el graf d'exemple del article de Shi et al. i es prova la consulta N a N en els casos especificats.

Objectius

Comprovar la funcionalitat de consulta N a N.

Entrada

Al fitxer `queryNtoN.in`.

Sortida

Al fitxer `queryNtoN.out`.

Resultat

Correcte.

14 queryByReference

Descripció

Els casos d'ús que es proven són:

- Consulta N a N (quan hi ha tipus intermig).
- Consulta N a N (quan no hi ha tipus intermig).

S'afegeix el graf d'exemple del article de Shi et al. i es prova la consulta per referència.

Objectius

Comprovar la funcionalitat de consulta per referència.

Entrada

Al fitxer `queryByReference.in`.

Sortida

Al fitxer `queryByReference.out`.

Resultat

Correcte.

15 queryAndResult

Descripció

Els casos d'ús que es proven són:

- Gestió de filtres al resultat.
- Ordenació del resultat.

S'afegeix el graf d'exemple del article de Shi et al. i es prova la consulta N a N. Amb el resultat obtingut s'apliquen els tipus possibles de filtres i s'ordena.

Objectius

Comprovar la funcionalitat del resultat.

Entrada

Al fitxer `queryAndResult.in`.

Sortida

Al fitxer `queryAndResult.out`.

Resultat

Correcte.