



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Estructuras Discretas



Proyecto 1: Relaciones

Arrieta Ocampo Braulio Enrique
López Santibáñez Jiménez Luis Gerardo
Silva García Carlos Sebastián

Fecha de entrega: 25 de abril de 2017

Relaciones entre parejas de elementos

Versión en línea de la aplicación:

<https://relations-project-stage.herokuapp.com/>

Ejemplos:

Relations 3001

Conjunto

1,2

Agregar



Reflexiva Irreflexiva Simétrica Asimétrica Antisimétrica Transitiva Relaciones

✓ ✗ ✓ ✗ ✗ ✓ [[{"1","2"}, {"2","1"}, {"1","1"}, {"2","2"}]]

Relations 3001

Conjunto

hola,adios,1,2

Agregar



Parejas

hola,1

adios,2

Reflexiva Irreflexiva Simétrica Asimétrica Antisimétrica Transitiva Relaciones

✗ ✓ ✗ ✓ ✓ ✓ [[{"hola","1"}, {"adios","2"}]]

✗ ✗ ✗ ✗ ✓ ✓ [[{"hola","1"}, {"adios","2"}, {"1","1"}, {"2","2"}]]

✗ ✗ ✗ ✗ ✗ ✗ [[{"hola","1"}, {"adios","2"}, {"1","1"}, {"2","2"}, {"1","2"}, {"2","1"}]]

✓ ✗ ✗ ✗ ✗ ✗ [[{"hola","1"}, {"adios","2"}, {"1","1"}, {"2","2"}, {"1","2"}, {"2","1"}, {"hola","hola"}, {"adios","adios"}]]

Documentación

Uso

En el campo de "Conjunto" deberá incluirse una lista de elementos (Números, letras o cadenas) separadas por coma. Los elementos son sensibles a mayúsculas por lo que es diferente el elemento "Hola" de "hola".

En los campos de elementos deberán ingresarse los elementos antes definidos separados por coma, en caso de no pertenecer alguno o ambos de los elementos se mostrará un mensaje de error para que sea corregido.

Para agregar más parejas deberá presionar el botón de "Agregar" el cual agregará más casillas para poder ingresar más relaciones. El número de relaciones no tiene límite.

Para ver las propiedades de las relaciones deberá presionar el botón "Enviar" el cual enviara los datos por medio de una petición a nuestro servidor y retornando los valores correspondientes que serán inmediatamente mostrados debajo.

Podrá repetir este proceso para N número de relaciones y/o conjuntos.

Para limpiar los datos y regresar al estado inicial es tan fácil como presionar el botón "Limpiar".

Descripción del proyecto

La carpeta ENV es un entorno virtual para poder correr el proyecto de manera local utilizando Heroku con el comando "heroku local" esto es necesario porque nuestro repositorio automáticamente es sincronizado desde el Dashboard de Heroku por lo cual los cambios se reflejan en cuestión de minutos en la aplicación web en línea.

Para correr el proyecto en local en cualquier computadora es necesario:

- tener Python 2 o 3 instalado
- tener pip instalado

y correr los comandos:

```
pip install requirements.txt
```

lo cual instalara lo necesario de "Flask" y posteriormente

```
python app.py
```

Lo cual inicializa la aplicación y la corre en "localhost" ahora solo debemos entrar a la dirección en consola y estamos listos para usar la aplicación en local.

Recordar que ya se encuentra en línea para su comodidad:

```
https://relations-project-stage.herokuapp.com/
```

Funcionamiento de los algoritmos en relaciones.py

La forma en cómo se verifican las propiedades para cada relación ingresada, es en base a una matriz de adyacencia, la cual es generada por la función 'createBinaryMatrix', que toma como parámetros un conjunto y las relaciones ingresadas, representando un 1 si están relacionados dos elementos y un 0 si no. Una vez generada la matriz, tenemos que:

- La reflexividad comprobada por la función 'isReflexive' verifica que en la diagonal principal todo elemento del conjunto se encuentre relacionado con sí mismo, o sea $[i][i] = 1$, para todo i .
- La irreflexividad es el caso contrario, la función 'isIrreflexive' verifica que ningún elemento del conjunto esté relacionado consigo mismo.
- Para comprobar si es simétrica, la función 'isSymmetric' verifica que para toda relación $[i][j]=1$ en la matriz, exista también $[j][i]=1$.
- Su caso contrario, para ver si la matriz es asimétrica, la función 'isAsimetric' verifica que para toda relación $[i][j]=1$, no exista $[j][i]=1$.
- Para ver si es antisimétrica, la función 'isAntisymmetric', comprueba que si $[i][j]=1$ y $[j][i]=1$, entonces $j=i$.
- Finalmente, la transitividad comprobada por la función 'isTransitive', verifica que si $[i][j]=1$ y $[j][k]=1$, entonces $[i][k]=1$, para todo i,j,k .