



Trabajo Practico Nº2

Rápidos y Furiosos

Materia: Programación III.

Profesor: Javier Marengo.

Patricia Bagnes.

Alumno: Ramos Aranibar Javier.



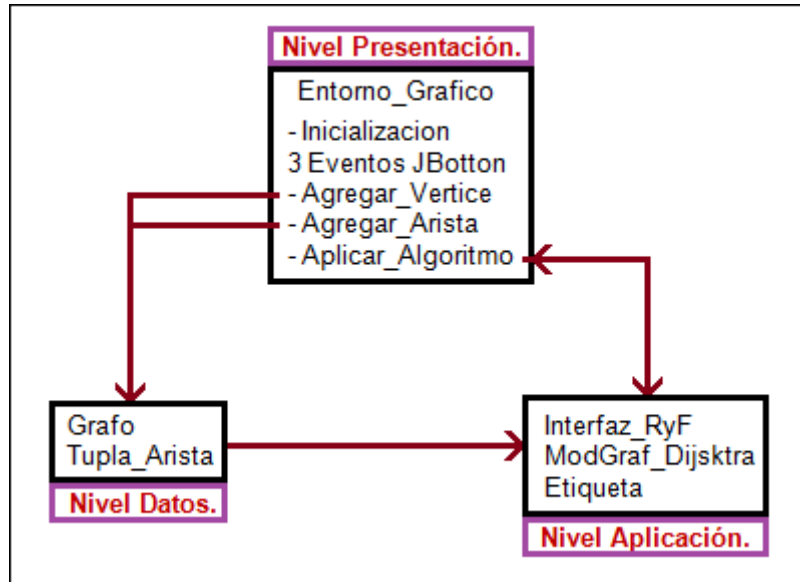
2017

Introducción

El presente trabajo practico tendrá como objetivo desarrollar la Aplicación Rapidos y Furiosos, el cual planifica los viajes en auto por las rutas del país, minimizando las distancias del viaje, pero con un límite de peajes a atravesar.

Diagrama de Clases

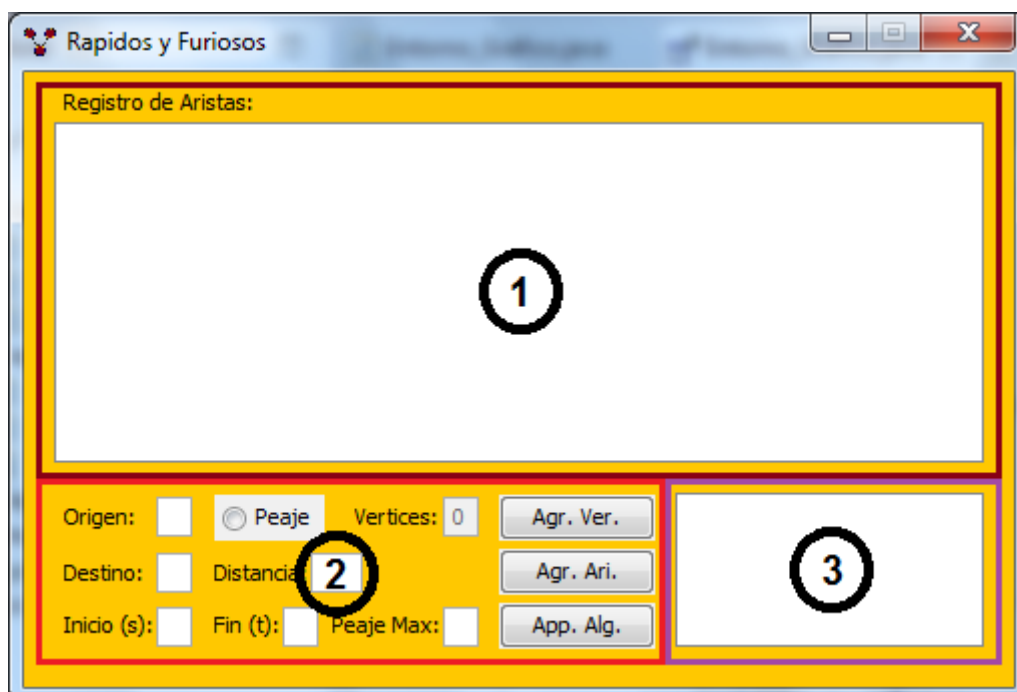
El siguiente diagrama, representa todas las clases utilizadas y su relación entre ellas. A continuación se explicara con más detalles, las rutas tomadas.



Podemos ver que nuestro código se divide en una arquitectura de 3 capas, los cuales interactúan entre sí; desde la Clase `Entorno_Grafico` a `Grafo` (Nivel de Datos) para casos de métodos simples como Agregar Vértices, Agregar Vértices, hacer Consultas del grafo, etc. hasta para realizar el proceso final, donde los algoritmos mas engorrosos se encuentran en otra clase aparte (Nivel Aplicación), con sus respectivos métodos.

Implementación

- Se opto por utilizar un único `JFrame`, ya que era innecesario implementar uno más. Además una aplicación que tiene mucho, quita atraktividad de quién lo va a usar.



Podemos ver en la captura, que la ventana se encuentra fraccionada en 3 partes.

- 1 Un JText cuya función es mostrar los registros de las aristas agregadas a mi grafo.
- 2 Una consola que posee 3 funciones: Agregar Vertice, Agregar Arista y Aplicar Algoritmo.
- 3 Un JText, que sirve de notificador. En ella se mostrara si se agrego una arista correctamente, si alguno de los datos ingresados es incorrecto, como así también mostrar el camino mínimo que se adapta a la consigna.

- Para evitar confusiones, se opto por omitir el 0 como nodo, es decir, se hizo un corrimiento a Nivel de Presentación, no si a Nivel de Datos, del rango de valores que se aceptan de vertices.

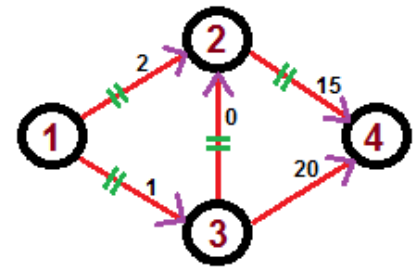
- Los JTexts tienen un Scroll asociado a cada uno, debido a que llegado un punto, los datos excederan en tamaño de este y perderemos visualización de los Registros/Solución.

Ejemplo

Registro de Aristas:

- Origen: 1 Distancia: 2 Destino: 2 Peaje: true
- Origen: 1 Distancia: 1 Destino: 3 Peaje: true
- Origen: 2 Distancia: 15 Destino: 4 Peaje: true
- Origen: 3 Distancia: 0 Destino: 2 Peaje: true
- Origen: 3 Distancia: 20 Destino: 4 Peaje: false

Origen: 3 ☐ Peaje Vertices: 4
Destino: 4 Distancia: 20
Inicio (s): Fin (t): Peaje Max: **Notificaciones:**



- Supongamos que tenemos el siguiente grafo.

- Donde las rayas verdes indican que la ruta posee peaje.

- Obj: se quiere llegar a 4 desde 1.

- Evaluemos los casos:

- Dado la cantidad de peajes, Tenemos 4 posibles casos.

- Peaje 0

Notificaciones:

- Distancia Min a recorrer: ∞ .
- Viajes a Destino: -

- Peaje 1

Notificaciones:

- Distancia Min a recorrer: 21.
- Viajes a Destino: -1-3-4-

- Peaje 2

Notificaciones:

- Distancia Min a recorrer: 17.
- Viajes a Destino: -1-2-4-

- Peaje 3

Notificaciones:

- Distancia Min a recorrer: 16.
- Viajes a Destino: -1-3-2-4-

Vemos que el resultado que tira mis 4 casos, cumple la consigna de tomar el camino más corto, teniendo en cuenta el límite/máximo de peajes a pagar.

Para este caso en particular, excediendo el peaje de 3, se obtendrá el mismo resultado.