Programación III - Universidad Nacional de General Sarmiento

Trabajo Práctico 2

**Docentes**:

* Patricia Bagnes
* Javier Marenco

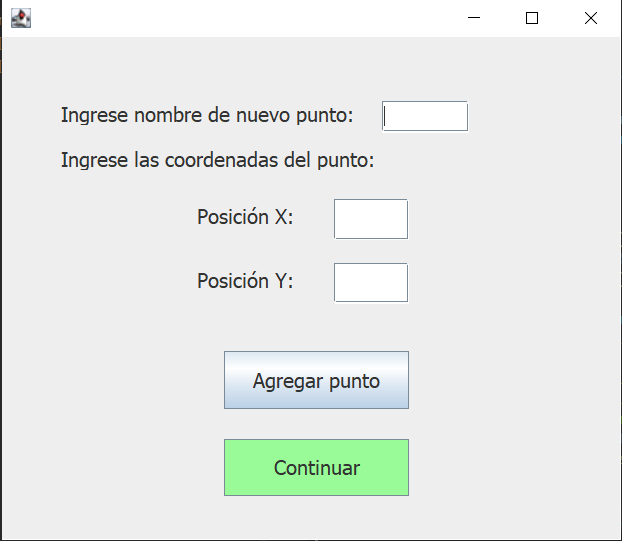
**Alumna**:

* Agustina Mendez
* Leandro Matias Avila

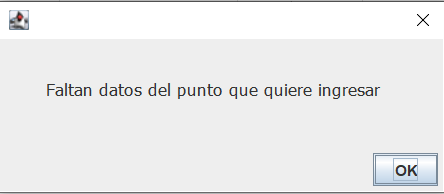
El objetivo del trabajo práctico es implementar una aplicación para realizar clustering utilizando el algoritmo propuesto por Charles Zahn en el artículo “Graph-theoretical methods for detecting and describing clusters” (IEEE Transactions on Computers 20-1, 1971). Dado un conjunto de puntos en el plano y un grafo completo con estos puntos y con pesos en las aristas iguales a la distancia euclídea entre los vértices, el algoritmo es el siguiente:

* Calcular un árbol generador mínimo del grafo.
* Eliminar del árbol las aristas cuyos pesos sean mayores que los pesos de sus aristas vecinas.

El sistema cuenta con una interfaz visual donde el usuario ingresa de manera manual: el nombre del vértice y sus coordenadas (x, y).



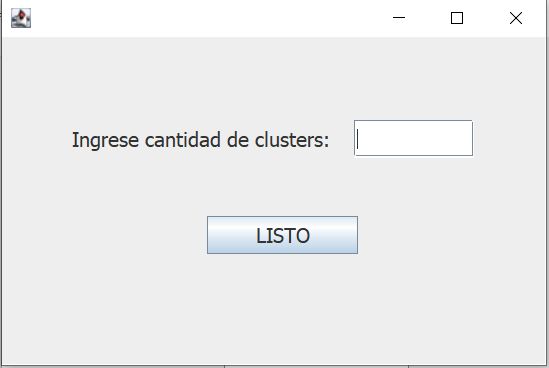
Si falta algún dato del punto ingresado, se muestra una ventana de dialogo para advertir al usuario.



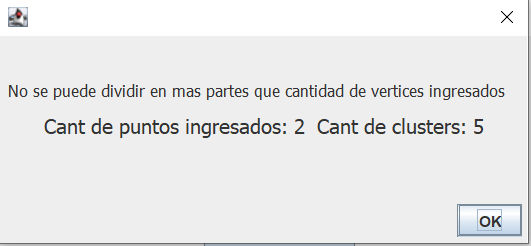
Una vez agregados los puntos, el usuario debe oprimir el boton **continuar**. Este botón llama al metodo del sistema **crearGrafoAGM()** que a su vez, llama al metodo **grafo.CrearGrafo()** que crea el grafo completo. A partir de ese grafo completo, se lo transforma en un AGM con la llamada al metodo **grafo.transformarArbolGeneradorMinimo()**.



El boton **continuar** tambien crea una nueva interfaz donde el usuario ingresara la cantidad de clusters en los que quiere dividir el grafo.



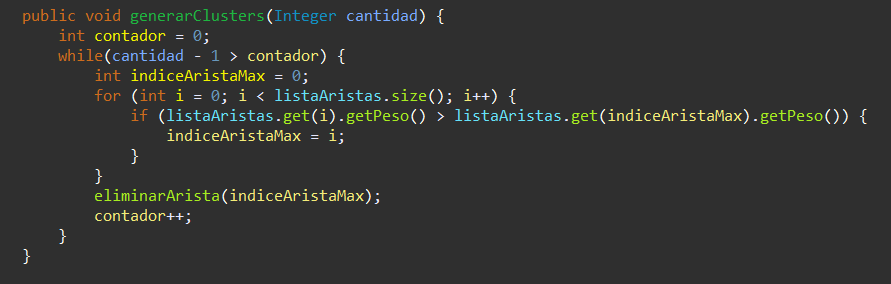
Al oprimir el boton **listo**, si la cantidad de divisiones que se quieren hacer es mayor a la cantidad de puntos ingresados, se mostrara una ventana de aviso para que se cambie el valor.



Si el valor ingresado es correcto, se hace la llamada al metodo **sistema.generarClusters().**



**sistema.generarClusters()** llama a **grafo.generarClusters(cantidad)**. Este metodo es el encargado de buscar las *cantidad – 1* aristas (siendo cantidad el número ingresado por el usuario) de mayor peso y eliminarlas hasta obtener la cantidad de componentes necesarias.

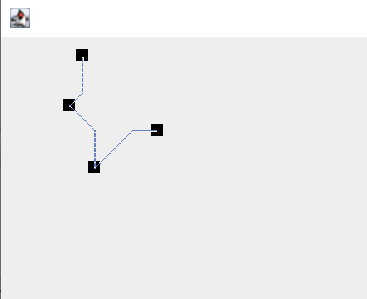


El boton **listo**, tambien abre la ventana donde se muestra el resultado final. Aca se puede observar gráficamente los puntos y las aristas de los clusters.

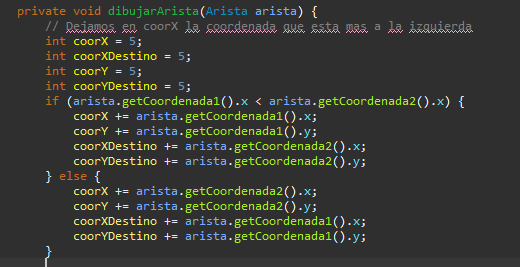
Para dibujar los vértices, utilizamos Panel de 10x10 y agregamos el nombre de cada vértice.



Para dibujar las aristas utilizamos JSeparator para formar una línea a partir de puntos. En un primer momento, para dibujar cada punto, nos movimos una posición en **x** y otra en **y**. Esto generaba que una vez alcanzado el valor de una de las coordenadas, se siguiera moviendo solo en **x** o en **y** hasta alcanzar la posición del segundo vértice.



Para que se vea mejor visualmente, se consideró obtener un porcentaje de movimiento para **x** y un porcentaje para **y**.





De esta manera logramos que las líneas se vean más parejas y que no haya “esquinas” que se puedan confundir con otros vértices.

