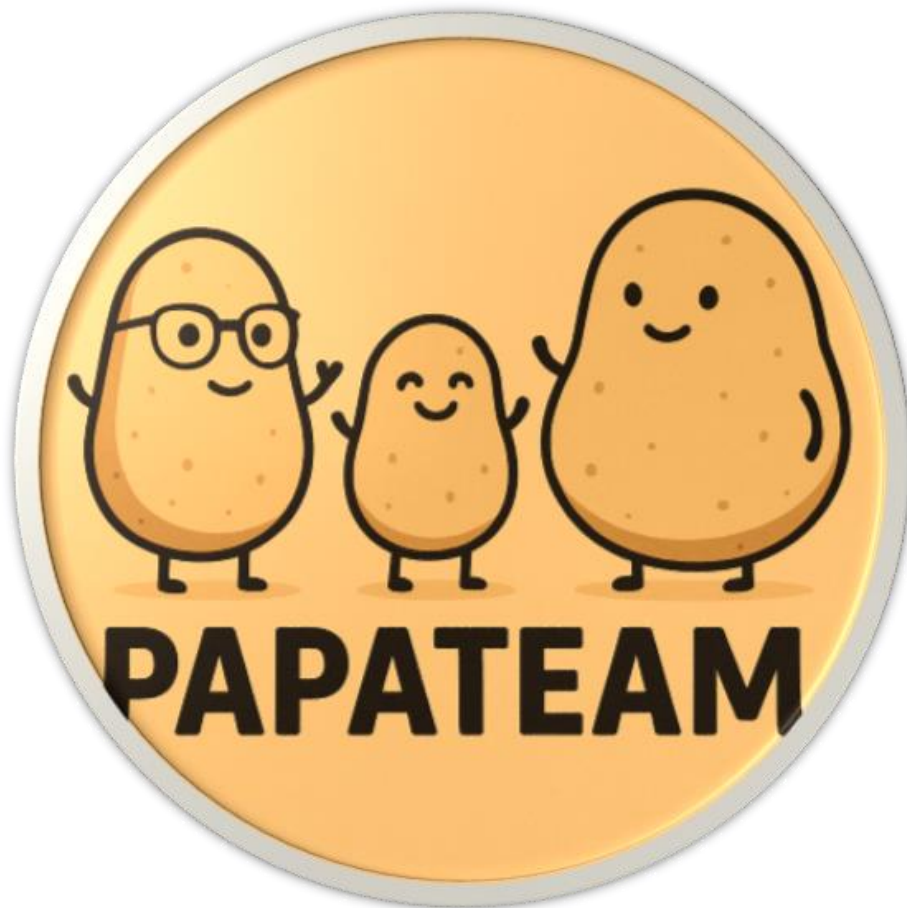


# ALGORITMOS Y ESTRUCTURAS DE DATOS II

INFORME TÉCNICO: TAREA N°9



- Agostini Santiago
- Pineda Nahuel
- Poviña Alejo

## CONTEXTO DEL PROYECTO

El sistema se diseñó en un escenario de logística urbana:

- Los depósitos representan centros de distribución.
- Los clientes son puntos de demanda.
- Los barrios se delimitan como polígonos.
- Los diagramas de Voronoi representan las áreas de influencia de cada depósito.
- Los clientes al agregarse al diagrama proceden a realizarse una línea al depósito correspondiente

El fin del trabajo integra la teoría y la práctica de la geometría computacional aplicando la triangulación de Delaunay y los diagramas de Voronoi en un minisistema modular en C++ orientado a la logística urbana.

## FUNCIONALIDADES DEL CODIGO

El código integra los siguientes algoritmos:

- Punto en polígono: determina si un cliente está dentro de un barrio (algoritmo ray casting).
- Intersección de segmentos: detecta cruces entre calles o tramos.
- Range query: filtra clientes dentro de un área delimitada.
- Nearest neighbor: conecta cada cliente con el depósito más cercano.
- Voronoi/Delaunay: construye regiones de influencia de depósitos y mallas trianguladas

## DISEÑO MODULAR DEL CODIGO

Cada algoritmo está encapsulado en un módulo independiente, lo que asegura mantenibilidad y extensibilidad.

- “Geometria.h”: definición de estructuras básicas (Point, Polygon) y función de distancia.
- “SegmentIntersect.h”: algoritmo de intersección de segmentos.
- “PointInPolygon.h”: verificación de pertenencia de punto en polígono.
- “RangeQuery.h”: consultas por rango (clientes dentro de un barrio).
- “NearestNeighbor.h”: búsqueda de vecino más cercano.
- “Voronoi.h”: construcción y visualización simple de Voronoi en SFML.
- main.cpp: integración de todas las funcionalidades y prueba de los algoritmos

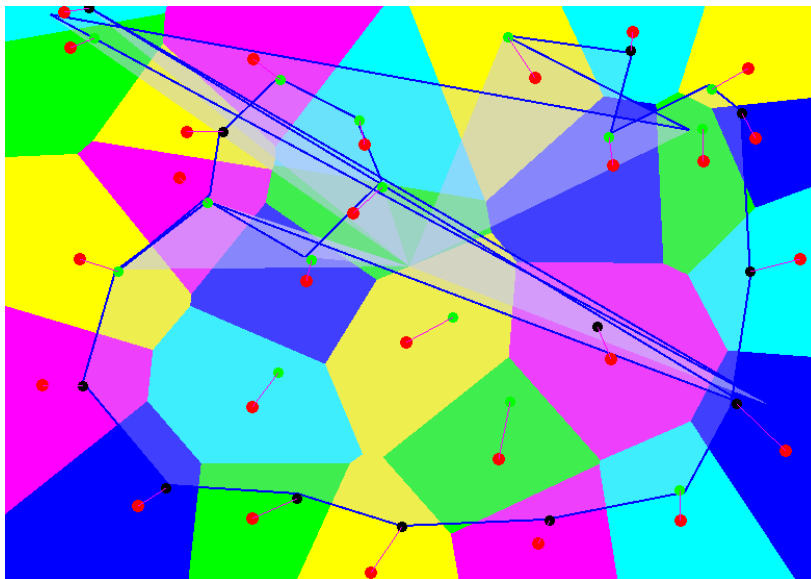
## EJEMPLIFICACION DEL CODIGO

- Se generan puntos aleatorios que representan depósitos y clientes.
- Se construye un KD-Tree para consultas de vecino más cercano y rango.
- Se calcula la triangulación de Delaunay y el diagrama de Voronoi.
- Se verifica el algoritmo de punto en polígono en un triángulo de la triangulación.
- Se prueba la intersección de segmentos con ejemplos simples.
- Los resultados se imprimen en consola, y opcionalmente se pueden visualizar con SFML

### Controles:

- Click izquierdo: agregar deposito
- Click derecho: agregar cliente
- Tecla P: comenzar/terminar poligono (barrio)
- Tecla R: resetear todo
- Cerrar ventana para salir

Dibuja poligono: click izquierdo para vertices, Enter para terminar.



## CONCLUSION

El trabajo consistió en el desarrollo de un mini sistema modular en C++ aplicado a la logística urbana, integrando algoritmos de geometría computacional como la asignación de clientes a depósitos, detección de pertenencia a barrios e intersección de segmentos y representación de zonas de influencia mediante diagramas de Voronoi. El diseño modular permitió realizar funciones eficaces, facilitando su reutilización y expansión. Junto al SFML mejoró la interacción y validación de los resultados. Este sistema resolvió los problemas del contexto logístico y facilitó la tarea del mismo.