



TALLER – ESTRUCTURA DE DATOS

Docente: PAUL LEGER.

Ayudante: DIONISIO OLIVARES.

Fecha de Inicio : 22 de Diciembre de 2020.

Fecha de Término : 12 de Enero de 2021, 23:59.

PLANTEAMIENTO DEL PROBLEMA

A lo largo del año 2020, nuestro país se ha visto afectado por el virus Covid-19. Lo cual ha unido a distintos profesionales como médicos, científicos e incluso informáticos a combatir este virus desde sus respectivas áreas.

Desde el comienzo de la pandemia, un grupo de programadores ha estado trabajando en un algoritmo para el ministerio de salud, el cual recoge información de cada comuna de Chile. El objetivo de este algoritmo es poder generar los balances del virus detalladamente para cada comuna del país en un tiempo aceptable. Además de poder obtener rápidamente el balance por sectores, ya que desde un principio los resultados del balance entregados a la comunidad tienen un cierto retardo desde la fecha original de los resultados.

Los algoritmos que actualmente han sido implementados por el grupo de programadores para poder generar los balances del virus tardan más de lo esperado, obteniendo una complejidad algorítmica de $O(n)$. Las estructuras de datos implementadas actualmente no permiten realizar consultas de contagiados por granularidad de un sector, comenzando desde todo Chile hasta una comuna en particular.

El Gobierno espera que la complejidad algorítmica del programa solicitado considere la granularidad de un sector y sea aceptable, para así poder obtener los balances del virus de una manera más rápida y permitirle a la comunidad y a los profesionales que combaten el virus puedan saber el estado de nuestro país en tiempo real. Es por esto por lo que el ministerio de salud ha solicitado a los alumnos de Estructura de Datos de la Universidad Católica del Norte implementar un nuevo algoritmo, el cual pueda satisfacer las necesidades del gobierno en un menor tiempo.

En cuanto al ingreso de los datos, el programa debe manejar lectura de archivos de texto, los cuales serán detallados a continuación.

- **Archivo de comunas:** El archivo comunas.txt contiene los datos de todas las comunas que el programa debe manejar. Cada línea del archivo contiene tres campos, el primero de ellos contiene el nombre de la comuna, mientras que los siguientes dos campos contienen un número decimal, el cual representa su ubicación en longitud y latitud respectivamente para cada comuna.

(Comuna, Longitud, Latitud)

- **Archivo de casos Covid-19:** El archivo casos.txt contiene los datos de casos confirmados para cada comuna. Cada línea del archivo está compuesta por el nombre de la comuna, el número de la población de la comuna y por último el número de casos confirmados del virus en dicha comuna.

(Comuna, Población, Casos Confirmados)

Nota 1: Para almacenar los datos de las comunas mencionados anteriormente se debe implementar una LinkedList y un QuadTree modificado. Un QuadTree es una estructura de datos, que se utiliza normalmente para almacenar y representar datos espaciales (por ejemplo: puntos en un determinado espacio con coordenadas en dos dimensiones).

A medida que se vayan ingresando los datos al programa, estos se deben insertar principalmente a la LinkedList, en donde cada Nodo de la LinkedList contendrá y representará a cada comuna ingresada con todos sus datos (Nombre, longitud, latitud, población, casos confirmados). Dicha lista enlazada debe estar ordenada con un orden de norte a sur (por su latitud). Gráficamente la lista sería de la siguiente manera: $A \rightarrow B \rightarrow C \rightarrow D$, en donde las comunas A y B; B y C; C y D “podrían” ser comunas contiguas o cercanas.

Una vez ingresadas todas las comunas a la LinkedList se debe construir el árbol que se utilizará, que en esta ocasión denominaremos “DualTree”, ya que cada nodo contendrá dos hijos y no cuatro como un QuadTree. Para este proceso debemos comenzar creando los nodos hojas del DualTree, los cuales en primera instancia serán almacenados en un puntero (arreglo) para ser referenciados más tarde. Es importante saber que un nodo del árbol es distinto al nodo de la LinkedList.

Cabe mencionar que, un Nodo del DualTree se identifica por las dos comunas a las que hace referencia, la longitud y latitud de un extremo superior, latitud y longitud de un extremo inferior, y la suma de casos confirmados en las comunas referenciadas.

Cuando se hayan creado todos los nodos hojas del DualTree, estos se deben unir con su nodo más cercano, es decir, cada nodo hoja del arreglo mencionado anteriormente se debe unir con otro para generar un nodo interno del árbol.

do en cuenta el ejemplo anterior, podemos generar dos sectores, los cuales
remos que son A con B, y C con D. Estos sectores se deben almacenar en un Puntero
o de sectores) con las referencias a las dos comunas agrupadas. El Nodo que
ente al sector debe contener la longitud y latitud de un extremo superior y la
ud y latitud de un extremo inferior, además de la suma de casos confirmados en las
as a las que referencia. Una vez generados todos los sectores referenciando a los
hojas, estos también se deben agrupar con sus sectores más cercanos (similar al
o anterior). Para poder obtener qué sector es el más cercano a otro se debe tener en
un punto medio entre los extremos (longitud y latitud) de cada sector. Este proceso
e repetir hasta unir los últimos dos sectores, quedando solamente la raíz del árbol
nciando a estos.

-
- Diagrama de flujo de datos que muestra la jerarquía de datos en un sistema de salud. Se muestran cuatro niveles: Nivel 0 (Raíz), Nivel 1, Nivel 2 y Nivel 3. Los datos fluyen desde la Raíz (Sector 3) hacia los Sectores inferiores (Sector 1 y 2), y luego hacia las Comunas (Comuna 1 y 2).
- Nivel 0 (Raíz):** Sector 3 (Sector 1 y 2). Contiene: Lng1; Lat1, Lng2; Lat2, Poblacion, Casos Confirmados.
- Nivel 1:** Sector inferior (Sector 1 y 2). Contiene: Lng1; Lat1, Lng2; Lat2, Poblacion, Casos Confirmados.
- Nivel 2:** Comuna 1 y Comuna 2. Contienen: Lng1; Lat1, Lng2; Lat2, Poblacion, Casos Confirmados.
- Nivel 3:** Hoja (Comuna 1 y Comuna 2). Contienen: Lng1; Lat1, Lng2; Lat2, Poblacion, Casos Confirmados.
- Las flechas indican el flujo de datos desde la Raíz hacia los Sectores inferiores, y luego hacia las Comunas.

- Las comunas corresponden a las hojas, mientras que el nodo denominado “Sector 3” corresponde a la raíz del DualTree.
- Para todos los nodos hoja (comunas 1, 2, 3 y 4) se cumple que, “ $\ln g1 = \ln g2$ ” y “ $lat1 = lat2$ ”.
- Para la raíz y nodos internos (Sector 1, 2 y 3) se cumple que, “ $\ln g1 \neq \ln g2$ ” y “ $lat1 \neq lat2$ ”.

- *Los casos confirmados del “Sector 1” corresponde a la suma de casos confirmados de la comuna 1 y casos de la comuna 2.*
- *Los casos confirmados del “Sector 3” (raíz) corresponde a la suma de casos en todas las comunas.*
- *La población se maneja de igual manera a los dos puntos anteriores,*

Nota 4: Debe tener cuidado al crear los sectores con las comunas contiguas o cercanas. Si crea un sector con dos comunas seleccionadas del arreglo (Comuna 1 y 2 por ejemplo) se recomienda identificarlas de alguna manera dentro del arreglo (ejemplo: boolean) que estas comunas ya han sido agrupadas en un sector, para que cuando cree otro sector estas comunas no se repitan en el sector nuevo. De igual manera cuando cree nuevos sectores con sectores inferiores, al crear un nuevo sector en otro nivel del árbol se recomienda que los sectores también puedan ser identificados dentro del programa que ya han sido ingresados en un sector superior y no se tomen en cuenta para cuando calcule el sector (o comuna) más cercano y que no se repita en otro sector superior o del mismo nivel.

- **REQUERIMIENTOS FUNCIONALES:**

- **RF1: Casos confirmados a nivel nacional.**

Se debe implementar un método dentro del DualTree, el cual permita saber la cantidad total de casos confirmados a nivel nacional.

- **RF2: Casos confirmados en un nivel del árbol.**

Dado un número ingresado por el usuario, el cual representa un nivel del árbol (altura), se debe desplegar por pantalla la distribución de casos confirmados separando por cada sector o comuna, dependiendo el nivel solicitado es el último nivel del árbol o no

Considerando el diagrama que se mostró anteriormente, para el nivel 1 se debería desplegar por pantalla el total de casos, luego los casos confirmados del “Sector 1” y por último los casos del “Sector 2”.

- **RF3: Casos confirmados entre puntos.**

Dado dos puntos ingresadas por el usuario como longitud1, latitud1, longitud2 y latitud2 se realizar una búsqueda dentro del árbol y desplegar por pantalla los casos confirmados de todas las comunas que estén en el rango solicitado.

- **RF4: Porcentaje de casos confirmados.**

- I. Porcentaje de casos confirmados:**

- Se debe desplegar por pantalla para cada comuna dentro del árbol el porcentaje de casos confirmados respecto a su población.

- II. Comunas en rango de porcentaje:**

- Dado un porcentaje a% y un porcentaje b% ingresado por el usuario, se debe realizar una búsqueda dentro del árbol y desplegar todas comunas que tienen un porcentaje de casos confirmados dentro del rango ingresado.

CONSIDERACIONES

- Implementación.
 - Se debe implementar un menú completo con los requerimientos solicitados.
 - Cuando se crea cada nivel nuevo se recomienda guardar los nodos que se van a agrupar primero en un arreglo e identificar cuando estos ya han sido agrupados con un nodo del mismo arreglo. Sin embargo, si considera otra manera más cómoda o fácil la puede implementar sin problemas.
 - Información sobre un QuadTree: <https://en.wikipedia.org/wiki/Quadtree>
- Entrega.
 - El taller debe ser realizado en grupos de 3 personas.
 - Seguir el “Código de Honor” establecido en el curso.
 - Los programas realizados deben desarrollarse en lenguaje C/C++ mediante el compilador
 - Cualquier entrega relacionada con código no debe tener problemas al compilar.
 - Se debe enviar el archivo main.cpp.
 - En caso de realizar la tarea en repl.it, debe probar su código en un compilador local en su computador antes de enviar para así evitar inconvenientes.
 - Las entregas deben ser subidas a la plataforma mediante un archivo ZIP con el formato *apellido1_apellido2_apellido3.zip*
 - En caso de dudas, realizar las consultas al correo del ayudante.
 - No se aceptarán entregas después de la fecha límite. Estas serán evaluadas con nota mínima 1,0.