

Método de la Ingeniería

Enunciado

Sitios turísticos

El departamento del Quindío es uno de los lugares turísticos más visitados en Colombia por su interminable lista de recursos naturales y de atractivos turísticos que motiva a pasar buenos momentos en familia. Además, la región reúne una serie de ventajas para hacer más grato su recorrido; entre otras cosas, por la cercanía entre los destinos y buenas y variadas alternativas para llegar a ellos. En el eje cafetero podemos encontrar una gran variedad de sitios aledaños como lo son: Salento, Córdoba, Calarca, entre otros.





Datos:

0. Genova
1. Pijao
2. Buenavista
3. Córdoba
4. La Tebaida
5. Armenia
6. Calarca
7. Montenegro
8. Quimbaya
9. Filandia
10. Salento

Sin embargo, se ha visto que en ocasiones los turistas desconocen las rutas y toman otras que hacen que gasten mucho más tiempo o que sencillamente no lleguen al destino que quieren, lo cual ha generado preocupación en la gobernación por el temor a que esto provoque que la cantidad de visitantes empiece a disminuir y se vea afectado en su economía.

Debido a esto, el gobernador del Quindío lo ha contratado a usted para desarrollar un programa que permita crear rutas para los viajeros con diferentes criterios, por ejemplo, la ruta más corta de un lugar a otro o rutas para llegar a un destino pasando por un lugar específico. Además, el programa debe permitir al usuario saber qué destinos se encuentran cerca a un lugar específico. Su programa debe estar en la condición de solucionar el problema con dos versiones de algoritmos de grafos vistos en clase.

Identificación del problema

El departamento de Quindío es muy conocido por su gran variedad de lugares turísticos a los cuales se puede visitar para deleitarse con su fauna y flora. Sin embargo, están experimentando una gran preocupación al saber que algunas personas que van de turistas no conocen las mejores rutas para llegar a dichos lugares. Entonces se requiere una solución para que dichas personas no gasten demasiado tiempo en rutas muy largas para llegar a un lugar específico, ya sea pasando primero por otro lugar o simplemente ir directo. Se deben proponer una solución con dos versiones de algoritmos de grafos vistos en clase

Requerimientos Funcionales

Nombre:	R. #1. Definir el tipo de algoritmo del grafo
Resumen:	El usuario deberá determinar el algoritmo de grafo que quiere para solucionar el problema
Entradas:	Algoritmo de grafo
Resultados:	Solución del problema con el algoritmo dado

Nombre:	R. #2. Permitir dar el recorrido más corto
Resumen:	Permitir dar el recorrido más corto para llegar a un destino
Entradas:	Destino y origen
Resultados:	Recorrido más corto

Nombre:	R. #.
Resumen:	
Entradas:	
Resultados:	

Recopilación de Información

fuelle: <https://www.turismoquindio.com/el-quindio/>

Recostado en la cordillera Central, entre cafetales rebosantes, está el departamento del Quindío, uno de los más pequeños de Colombia, cuya extensión reducida es totalmente opuesta a la interminable lista de recursos naturales y de atractivos turísticos que motivan su frecuente visita. Además, la región reúne una serie de ventajas para hacer más grato su recorrido; entre otras cosas, por la cercanía entre los destinos y las buenas y variadas alternativas para llegar a ellos.

Incluía Armenia, la capital, al departamento lo conforman 12 municipios.

A todos se llega, para todos hay camino y en todos hay atrayentes para detenerse, contemplar y disfrutar. Aunque con algunos rasgos distintos, en los pueblos se notan muchas coincidencias, como los arbustos colmados de café o la guadua emergiendo de la tierra o transformada en arte y vivienda.

El Departamento del Quindio, hace parte del territorio colombiano, declarado patrimonio por la UNESCO, como Paisaje Cultural Cafetero.

ECOTURISMO Y PAISAJE

Surge en el Quindío un vistoso panorama de cafetales, flores, platanales y guaduales. Más de 2.000 especies vegetales y un conjunto de tierras fértiles, donde se crea un ambiente multicolor con predominio de verdes y el abrigo de una temperatura de 20 grados C en la mayor parte del departamento, excepción hecha de escenarios como el valle de Cocora y el parque Los Nevados.

Entonces este destino, que ocupa menos del 1 % de toda la superficie colombiana, abre caminos por entre sus pródigos cultivos y extensas reservas naturales, para que así los ecologistas y los amantes del paisaje tengan un espacio de observación y conocimiento frente a los maravillosos recursos con los que cuenta la geografía quindiana.

Por allí vuelan más de 500 especies de aves para su avistamiento, observación y registro fotográfico sea una de las actividades más placenteras durante las caminatas por varias reservas como Bremen en Filandia, El Ocaso en Quimbaya, El Jardín en Génova y el mismo valle de Cocora en Salento.

Grafo

fuelle: https://es.wikipedia.org/wiki/Teor%C3%ADa_de_grafos

La teoría de grafos, también llamada teoría de gráficas, es una rama de las matemáticas y las ciencias de la computación que estudia las propiedades de los grafos. Los grafos no deben ser confundidos con las gráficas, que es un término muy amplio. Formalmente, un grafo $G = (V, E)$ es una pareja ordenada en la que V es un conjunto no vacío de vértices y E es un conjunto de aristas. Donde E consta de pares no ordenados de vértices, tales como que (X, Y) pertenecen a E entonces se dice que X y Y son adyacentes; y en el grafo se representa mediante una línea no orientada que una dichos vértices. Un grafo permite representar relaciones binarias entre elementos de un conjunto.

Se puede formular un grafo donde cada nodo representa un lugar turístico del departamento de Quindío, cada vértice representa la unión (camino entre un lugar y otro) y el peso de cada uno de los vértices como la distancia (en KM) entre un lugar A y un lugar B.

Puesto que el problema representa lugares turísticos del departamento de Quindío es necesario listar todos los lugares a los que se pueden llegar.

Datos:

0. Genova
1. Pijao
2. Buenavista
3. Córdoba
4. La Tebaida
5. Armenia
6. Calarca
7. Montenegro
8. Quimbaya
9. Filandia
10. Salento

Ya conocidos estos datos, para que el problema sea lo más acertados posible y cercano a la realidad, se necesita conocer ahora las distancia (en kilómetros) entre cada lugar turístico del Departamento de Quindío.

1. Genova - Pijao 10.1 KM
2. Genova - Buenavista 14.4KM
3. Pijao - Cordoba 12.0 KM
4. Buenavista - Cordoba 20 KM
5. Cordoba - Calarca 30 KM
6. Calarca - Montenegro 19 KM
7. Calarca - La Tebaida 23 KM
8. La Tebaida - Montenegro 19 KM
9. Montenegro - Quimbaya 12 KM
10. Quimbaya - Filandia 19 KM
11. Filandia - Salento 23 KM

Búsqueda de soluciones creativas

Afrontaremos el problema de las sitios turísticos con diferentes enfoques

Alternativa 1: En este enfoque, iniciaremos con una interfaz gráfica estéticamente agradable a la vista del usuario en donde le pediremos cierta información para guardar como lo son su nombre, el lugar de origen y al que desea llegar, después haremos una especie de simulación en donde pintaremos de color por cada ciudad que esté pasando según el tipo de algoritmo que haya escogido para la ruta, tendremos una base de datos la cual nos ayudará a recopilar la información de las ciudades y las distancias entre ellas, luego de que haya terminado la simulación, buscaremos estimar el tiempo en el que tardará en realizar todo el viaje. Con esta alternativa utilizaremos el Algoritmo de Prim y Dijkstra

Mockup

Planeación de viaje

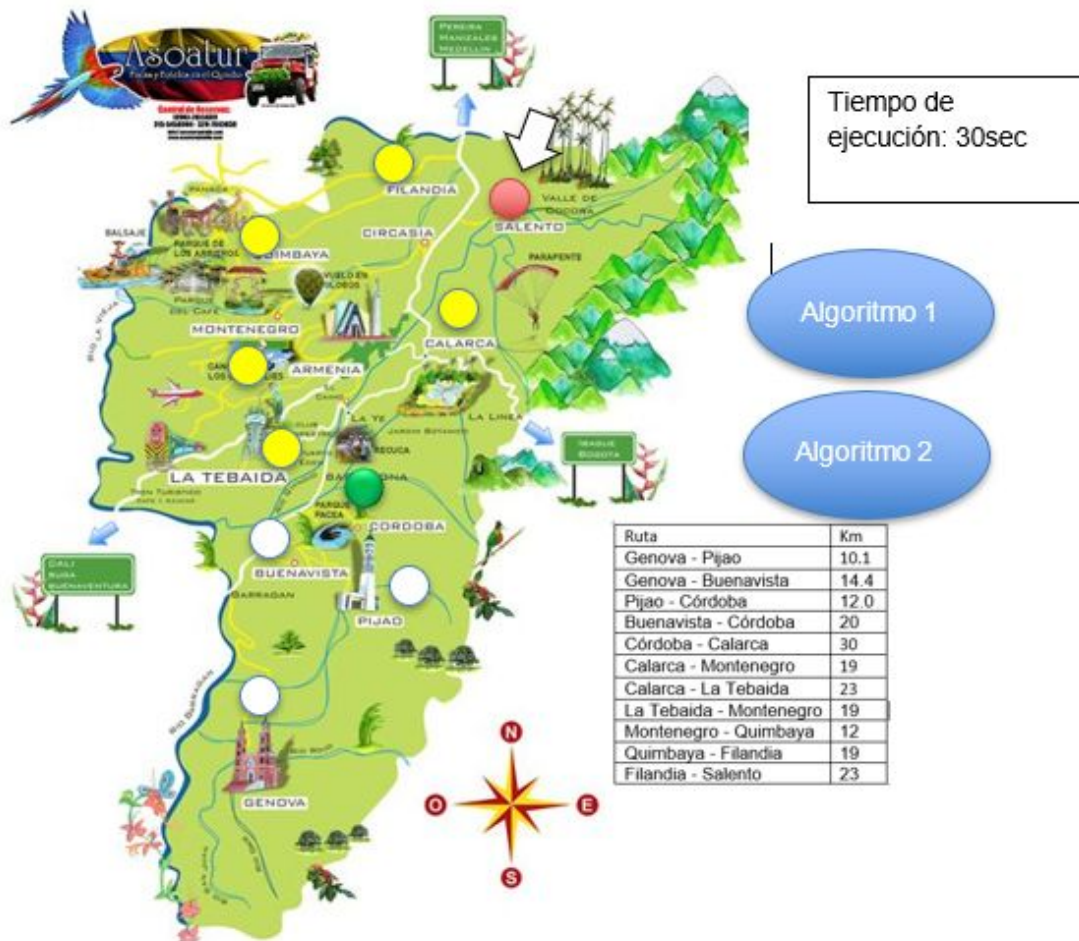
Nombre: Juan Hernandez
Lugar que desea ir: Córdoba
Lugar de origen: Salento
Ruta que desea: Corta

VISUALIZACIÓN



Alternativa 2: En este enfoque iniciaremos con una interfaz gráfica estéticamente agradable a la vista del usuario en donde se mostrará el mapa con todos los sitios turísticos como vértices y además estilo botón, en donde lo llevará a otra escena en donde se le brinde información acerca del lugar si es presionado, luego otro botón para planear ruta y permita seleccionar los vértices como inicio y fin para empezar la ruta, se colocará de rojo el vértice inicial, de verde el vértice final, luego saldrán dos botones para escoger el algoritmo que desea utilizar y por último el tipo de ruta que desea emplear, luego de que todo esto se haya seleccionado, se desaparecerán los vértices por los cuales no se recorre y pintará por los que sí de color amarillo y por último dará el tiempo que le costó al algoritmo emplearse.

Mockup



Alternativa 3: En este enfoque queremos implementar una interfaz simple, pero interactiva con el usuario que podamos seleccionar de check box el lugar donde queremos iniciar y al que queremos llegar, también seleccionar el tipo de ruta y cuando todo sea seleccionado saldrá un imagen de un señor que recorre la ruta que el algoritmo recomienda, también al poner el mouse encima del vértice saldrá una pequeña descripción acerca del lugar. Otra cosa es que se permitirá visualizar las tablas de adyacencia y incidencia.

Mockup



Alternativa 4: Programar el programa por consola para que los dispositivos no gasten tantos recursos con una interfaz gráfica.

Alternativa 5: Hacer el programa utilizando grafos implementados con una matriz de adyacencia para tener un mejor control de los datos y hacer las operaciones de una manera más fácil. Esta forma es muy conveniente cuando la cantidad de datos o información es mucha.

Alternativa 6: Hacer el programa utilizando grafos implementados con una lista de adyacencia para tener un mejor control de los datos y hacer las operaciones de una manera más fácil. Esta forma es muy conveniente si la cantidad de información no es mucha.

Alternativa 7: Hacer el programa utilizando ArrayList para poder agregar datos más fácilmente y no tener que gastar mucho tiempo. Aquí cada lugar tendría su propia ArrayList con las carreteras o ciudades que lo conectan.

Alternativa 8: Hacer el programa incluyendo rutas de todo el país para llegar al departamento del Quindío desde otro de Colombia.

Diseños preliminares

En este índice vamos a descartar algunas ideas que pueden no ser factibles o incluso inútiles.

La primera que vamos a descartar es la alternativa 4 porque el cliente nos está pidiendo que el programa tenga una interfaz y, además, si el programa fuera por consola los usuarios tendrían muchos problemas para utilizarlos por el desconocimiento acerca del funcionamiento.

La segunda que vamos a descartar es la alternativa 8 puesto que el cliente pide un programa sólo para el departamento, por lo que hacer el programa con rutas de todo el país sería una pérdida de tiempo y de recursos.

No vamos a descartar más alternativas, por lo tanto vamos a pasar la evaluación y selección.

Evaluación y selección de la mejor solución

Para evitar tener criterios que no apliquen en algunas alternativas vamos a hacer una evaluación para las alternativas que tengan que ver principalmente con interfaces gráficas y otra para las demás.

Evaluación de interfaces:

- Nivel de dificultad de uso para el usuario:
 1. Imposible de usar, para nada es intuitiva.
 2. Difícil de usar, poco intuitiva.
 3. Fácil de usar, intuitiva.

- Cantidad de recursos usados del dispositivo:
 1. Muchos recursos necesarios, puede que el dispositivo se ponga lento.
 2. Pocos recursos necesarios, el programa se puede correr en cualquier dispositivo.
- Nivel de completitud:
 1. Nada completa, no cumple con los requisitos del cliente.
 2. Completa, sin embargo, tiene más de lo que se pide y se sale del presupuesto.
 3. Completa, tiene exactamente lo que pide el cliente.

# de alternativa	Dificultad	Cantidad de recursos	Completitud	Total
1	3	2	3	8
2	2	1	2	5
3	3	1	2	6

Evaluación de algoritmos:

- Acoplamiento y desacoplamiento:
 1. Muy acoplado, difícil de reutilizar código en caso de que el cliente pida más cosas.
 2. Poco acoplado, fácil de reutilizar código.
- Peso del programa:
 1. Muy pesado, demasiados recursos utilizados.
 2. Pesado, puede ser por el uso de librerías importadas.
 3. Liviano, no hace que el programa pese.
- Tiempo de implementación:
 1. Bastante, se puede tardar más de una hora para hacer una función.
 2. Poco, se implementa muy rápido y fácil.

# de alternativa	Acoplamiento	Peso	Tiempo de implementación	Total
5	2	3	2	7
6	2	3	2	7
7	1	2	2	5

Después de la evaluación vemos que la alternativa 1 es la mejor para las interfaces, mientras que, por otro lado, vamos a escoger 2 alternativas de la parte de codificación puesto que así lo pide el cliente. Estas últimas van a ser la alternativa 5 y 6.