**Universidad Peruana de Ciencias Aplicadas**



Competencia de Pensamiento Innovador

CURSO DE Complejidad Algoritmica - CC182

Carrera de Ciencias de la Computación

Sección: CC41

Alumno: Juan Sebastian Osorio Quiroz

Código: u201713107

Julio 2022

**CONTENIDO**

1. Uso del conocimiento
2. Identificación del problema
3. Idea de solución al problema
4. Propuesta
5. **USO DEL CONOCIMIENTO**

**Para la realización de mi proyecto final utilicé conocimientos previos, además los aprendidos a lo largo del curso, de los cuales usé listas de adyacencia para la creación del grafo y tener un mejor manejo , el algoritmo de Dijkstra para la obtención del camino más optimo , también use Perlin Noise para generar pesos de las aristas de los grafos con un ruido suave y no sea tan distanciado los pesos , y como conocimientos previos seria la lectura de archivos esto lo realice para la lectura de mi lista de adyacencia.**

**Con todos los conocimientos obtenidos los aplique para la realización del mapa que su función era encontrar la ruta mas corta de un punto a otro dependiendo si tenia trafico bajo , medio o alto, primero se creo la lista de adyacencia para luego con perlin noise darles peso según el grado de tráfico , entonces una vez obtenido la lista con los pesos respectivos se utilizo el algoritmo de Dijkstra para la obtención del camino mas corto y para terminar se utilizó la librería flask para la parte visual donde se mostraría el mapa y sus rutas mas cortas.**

1. **IDENTIFICA EL PROBLEMA**

**Para este trabajo el problema inicial fue elegir la cuidad en la cual nos basamos para nuestro mapa, en conjunto se opto por Washington DC ya que es una de las cuidades mas ordenadas y planificadas del mundo, lo que no permitiría al momento de graficarlo una representación lo más parecida posible ya que el objetivo planteado en el proyecto era lograr las rutas más cortas.**

**Se decidió por utilizar las listas de adyacencia ya que optimizarían el programa al tener un buen uso de memoria y nuestro número de aristas es menor a O (), además se puede representar con los pesos de las aristas que fueron obtenidos con perlin noise y con un fácil uso en el algoritmo de Dijkstra.**

1. **IDEAR**

**En nuestra reunión de grupo nos centramos en elegir la ciudad para la realización esto nos facilitaría a la hora de crear la lista de adyacencia, para su creación nos ayudamos de la herramienta de Google maps de “Crea tu mapa” con los cuales se puede crear puntos de referencia que sirven como nodos, además Google maps nos brinda también las coordenadas de nuestros nodos**

**Con la data obtenida se pudo crear una estructura de cómo se vería el grafo, al tener los 1500 nodos se paso a crear la lista y con ayuda del perlin noise se pasó a otorgarles su respectivo peso, ya listo la lista se procedió a utilizar el algoritmo de Dijkstra con la ruta más corta.**

**Ya en la parte final del proyecto toda nuestra data se pasó a visual code donde se graficó el mapa de Washington.**

1. **PROPUESTA**

**Para la metodología de trabajo de nuestro grupo optamos por trabajar primero en collab y subir nuestros avanzases en Github en los hitos correspondientes, a lo largo del trabajo nos repartimos en partes equitativas como la creación de los pesos, la obtención de los paths con el algoritmo de Dijkstra y la parte visual por mencionar algunas tareas realizadas.**

**Para la parte final del proyecto enfocada en lo visual optamos por pasarnos a visual studio code ya que importando algunas librerías nos facilitaba este proceso y no se hacía muy engorroso, logrando con éxito la visualización del mapa y sus rutas.**

**Para finalizar, en el transcurso del trabajo surgieron algunas dificultades al programar ciertas partes del código, como la obtención de los paths y el peso de la lista, pero con el esfuerzo del grupo se logró solucionar satisfactoriamente.**