DOCUMENTO TÉCNICO

# Requisitos

## Especificación

#RESUMEN DEL PROBLEMA (opcional)

Implementar la toma de vecinos en la matriz de adyacencia

### Entrada:

### Matriz de adyacencia

### Salida:

# vértice con su correspondiente vecino

# Diseño

## Estrategia

### Para implementar la toma de vecinos en la matriz de adyacencia se siguieron 4 pasos:

### Verificar si el vértice existe en el grafo es decir si esta presente en el diccionario (encoder)

### Si el vértice existe, obtener el índice del vértice en la matriz de adyacencia

### Recorrer la fila del índice obtenido en el paso b

### Por cada valor de 1 que se encuentre, significa que es un vecino, por lo tanto, se agregara a la lista de vecinos

### Finalmente se retorna la lista de vecinos

### Para buscar un prototipo funcional en un escenario en la vida real, tome los vértices como ubicaciones y las relaciones como conexiones entre las ciudades. Si queremos realizar un viaje de una ciudad a hacia una ciudad b, realizamos el corrido bfs y ya realizado este tomamos el camino que se generó en estas 2 ciudades.

### #Estructuras de datos

Conjuntos, Diccionarios y Listas

#ALGORITMO

Programa adjunto

## Casos de prueba

|  |  |  |
| --- | --- | --- |
| **Entrada** | **Justificación** | **Salida** |
| Ubicacion inicial =1  Ubicacion final=8 | Caso genérico | Camino: [1, 2, 4, 6, 8] |
|  |  |  |
|  |  |  |
|  |  |  |

# Análisis

# #Protoripo funcional

# Texto Descripción generada automáticamente

# 

## Temporal

La complejidad de este código es aproximadamente de orden de E + orden de V^2 en el peor de los casos debido a la construcción de la matriz de adyacencias. Si se habla de lista de adyacencia en el peor de los casos seria de orden de V^2 , donde esto significaría que cada vertice esta relacionado por lo menos una vez con los demás, incluyéndose a si mismo

# 

# Código

*Al igual que el punto anterior pueden guiarse con este ejemplo*

## Documentación

Dentro del código.

## Fuentes

*Las fuentes son el código sobre el cual escribieron el documento*

/arena\_1\_a