

# OBSERVACIONES DE LA PRÁCTICA

Mateo Oviedo Reyes 202028312

Angie Catalina Campos Perilla 202115094

## Preguntas de análisis

### a) ¿Qué instrucción se usa para cambiar el límite de recursión de Python?

Para cambiar el límite de recursión de Python se usa la instrucción **sys.setrecursionlimit(n)** donde n hace referencia al límite de recursión que desees.

### b) ¿Por qué considera que se debe hacer este cambio?

Este cambio se realiza siempre y cuando sea necesario y/o se requiera, dado que pueden existir casos donde el programa arroje un mensaje de error por exceder el límite de recursión previamente establecido y por esto se requiere ampliar este límite.

### c) ¿Cuál es el valor inicial que tiene Python como límite de recursión?

Este valor inicial para el límite de recursión es de **1000**.

### d) ¿Qué relación creen que existe entre el número de vértices, arcos y el tiempo que toma la operación 4?

A medida que aumenta el número de vértices y arcos, también aumenta el tiempo de ejecución dado que la operación 4, lo que hace es recorrer todos los vértices del grafo desde la estación dada por parámetro.

**Archivo:** bus\_routes\_50.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 74
Numero de arcos: 73
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 28.796000000000042

```

**Archivo:** bus\_routes\_150.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 146
Numero de arcos: 146
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 39.434999999999995

```

**Archivo:** bus\_routes\_300.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 295
Numero de arcos: 382
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 74.415

```

**Archivo:** bus\_routes\_1000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 984
Numero de arcos: 1633
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 366.82000000000005

```

**Archivo:** bus\_routes\_2000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 1954
Numero de arcos: 3560
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 1325.5359999999998

```

**Archivo:** bus\_routes\_3000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 2922
Numero de arcos: 5773
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 2225.064

```

**Archivo:** bus\_routes\_7000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 6829
Numero de arcos: 15334
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 8413.125

```

**Archivo:** bus\_routes\_10000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 9767
Numero de arcos: 22758
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 17948.506999999998

```

**Archivo:** bus\_routes\_14000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>2

Cargando información de transporte de singapur ....
Numero de vertices: 13535
Numero de arcos: 32270
El limite de recursion actual: 1048576

```

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>4
Estación Base: BusStopCode-ServiceNo (Ej: 75009-10): 75009-10
El tiempo de ejecución es: 33066.22900000001

```

- e) ¿El grafo definido es denso o disperso?, ¿El grafo es dirigido o no dirigido?,  
¿El grafo está fuertemente conectado?

El grafo es **disperso**, para determinarlo se utilizó la siguiente fórmula para determinar su densidad:  $a \div (v(v-1))$ , donde  $a$  es el número de arcos y  $v$  es el número de vértices; donde al realizar los respectivos cálculos, todos dieron un valor inferior a 0.3, lo cual da a entender que es disperso. Además, se trata de un grafo **dirigido**, dado que no recorre todo el grafo, así mismo, tiene la posibilidad de encontrar el camino más corto por medio del algoritmo de ruta mínima de Dijkstra. De igual manera, el grafo **no está fuertemente conectado** puesto que tiene una cantidad reducida de componentes conectados.

**Archivo:** bus\_routes\_50.csv



```

*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 59
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.400000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.20000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.40000000000000036}
{'vertexA': '85049-10', 'vertexB': '85039-10', 'weight': 0.2999999999999998}
{'vertexA': '85039-10', 'vertexB': '85029-10', 'weight': 0.20000000000000018}
{'vertexA': '85029-10', 'vertexB': '85019-10', 'weight': 0.20000000000000018}
{'vertexA': '85019-10', 'vertexB': '94079-10', 'weight': 0.2999999999999998}
{'vertexA': '94079-10', 'vertexB': '94069-10', 'weight': 0.5999999999999996}
{'vertexA': '94069-10', 'vertexB': '94059-10', 'weight': 0.30000000000000007}
{'vertexA': '94059-10', 'vertexB': '94049-10', 'weight': 0.39999999999999947}
{'vertexA': '94049-10', 'vertexB': '94089-10', 'weight': 0.20000000000000018}
{'vertexA': '94089-10', 'vertexB': '94039-10', 'weight': 0.2999999999999998}
{'vertexA': '94039-10', 'vertexB': '94029-10', 'weight': 0.40000000000000036}
{'vertexA': '94029-10', 'vertexB': '94019-10', 'weight': 0.20000000000000018}
{'vertexA': '94019-10', 'vertexB': '93099-10', 'weight': 0.40000000000000036}
{'vertexA': '93099-10', 'vertexB': '93089-10', 'weight': 0.5}
{'vertexA': '93089-10', 'vertexB': '93079-10', 'weight': 0.29999999999999983}
{'vertexA': '93079-10', 'vertexB': '93069-10', 'weight': 0.30000000000000007}
{'vertexA': '93069-10', 'vertexB': '93059-10', 'weight': 0.40000000000000036}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999983}
{'vertexA': '92159-10', 'vertexB': '92149-10', 'weight': 0.30000000000000007}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999983}
{'vertexA': '92139-10', 'vertexB': '92129-10', 'weight': 0.5}
{'vertexA': '92129-10', 'vertexB': '92119-10', 'weight': 0.5}
{'vertexA': '92119-10', 'vertexB': '92109-10', 'weight': 0.20000000000000018}
{'vertexA': '92109-10', 'vertexB': '92099-10', 'weight': 0.29999999999999983}
{'vertexA': '92099-10', 'vertexB': '92161-10', 'weight': 0.5}
{'vertexA': '92161-10', 'vertexB': '82071-10', 'weight': 0.40000000000000036}
{'vertexA': '82071-10', 'vertexB': '81199-10', 'weight': 0.8000000000000007}
{'vertexA': '81199-10', 'vertexB': '81189-10', 'weight': 0.29999999999999983}
{'vertexA': '81189-10', 'vertexB': '81179-10', 'weight': 0.5}
{'vertexA': '81179-10', 'vertexB': '80141-10', 'weight': 0.60000000000000014}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80169-10', 'weight': 1.3999999999999996}
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.70000000000000028}
{'vertexA': '80159-10', 'vertexB': '02119-10', 'weight': 0.5}
{'vertexA': '02119-10', 'vertexB': '03019-10', 'weight': 1.0}
{'vertexA': '03019-10', 'vertexB': '03059-10', 'weight': 0.299999999999999716}
{'vertexA': '03059-10', 'vertexB': '03129-10', 'weight': 0.400000000000000213}
{'vertexA': '03129-10', 'vertexB': '03217-10', 'weight': 0.39999999999999986}
{'vertexA': '03217-10', 'vertexB': '05649-10', 'weight': 0.90000000000000021}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.299999999999999716}
{'vertexA': '14061-10', 'vertexB': '14071-10', 'weight': 0.5}
{'vertexA': '14071-10', 'vertexB': '14081-10', 'weight': 0.400000000000000213}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.39999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.8000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.30000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.39999999999999986}
{'vertexA': '14131-10', 'vertexB': '14151-10', 'weight': 0.5}
{'vertexA': '14151-10', 'vertexB': '14161-10', 'weight': 0.400000000000000213}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.39999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.30000000000000007}
El tiempo de ejecución es: 0.5470000000000197

```

**Archivo:** bus\_routes\_150.csv

```

*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 59
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.400000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.20000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.40000000000000036}
{'vertexA': '85049-10', 'vertexB': '85039-10', 'weight': 0.2999999999999998}
{'vertexA': '85039-10', 'vertexB': '85029-10', 'weight': 0.20000000000000018}
{'vertexA': '85029-10', 'vertexB': '85019-10', 'weight': 0.20000000000000018}
{'vertexA': '85019-10', 'vertexB': '94079-10', 'weight': 0.2999999999999998}
{'vertexA': '94079-10', 'vertexB': '94069-10', 'weight': 0.5999999999999996}
{'vertexA': '94069-10', 'vertexB': '94059-10', 'weight': 0.30000000000000007}
{'vertexA': '94059-10', 'vertexB': '94049-10', 'weight': 0.39999999999999947}
{'vertexA': '94049-10', 'vertexB': '94089-10', 'weight': 0.20000000000000018}
{'vertexA': '94089-10', 'vertexB': '94039-10', 'weight': 0.2999999999999998}
{'vertexA': '94039-10', 'vertexB': '94029-10', 'weight': 0.40000000000000036}
{'vertexA': '94029-10', 'vertexB': '94019-10', 'weight': 0.20000000000000018}
{'vertexA': '94019-10', 'vertexB': '93099-10', 'weight': 0.40000000000000036}
{'vertexA': '93099-10', 'vertexB': '93089-10', 'weight': 0.5}
{'vertexA': '93089-10', 'vertexB': '93079-10', 'weight': 0.29999999999999983}
{'vertexA': '93079-10', 'vertexB': '93069-10', 'weight': 0.30000000000000007}
{'vertexA': '93069-10', 'vertexB': '93059-10', 'weight': 0.40000000000000036}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999983}
{'vertexA': '92159-10', 'vertexB': '92149-10', 'weight': 0.30000000000000007}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999983}
{'vertexA': '92139-10', 'vertexB': '92129-10', 'weight': 0.5}
{'vertexA': '92129-10', 'vertexB': '92119-10', 'weight': 0.5}
{'vertexA': '92119-10', 'vertexB': '92109-10', 'weight': 0.200000000000000107}
{'vertexA': '92109-10', 'vertexB': '92099-10', 'weight': 0.29999999999999983}
{'vertexA': '92099-10', 'vertexB': '92161-10', 'weight': 0.5}
{'vertexA': '92161-10', 'vertexB': '82071-10', 'weight': 0.40000000000000036}
{'vertexA': '82071-10', 'vertexB': '81199-10', 'weight': 0.80000000000000007}
{'vertexA': '81199-10', 'vertexB': '81189-10', 'weight': 0.29999999999999983}
{'vertexA': '81189-10', 'vertexB': '81179-10', 'weight': 0.5}
{'vertexA': '81179-10', 'vertexB': '80141-10', 'weight': 0.60000000000000014}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80169-10', 'weight': 1.3999999999999986}
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.70000000000000028}
{'vertexA': '80159-10', 'vertexB': '02119-10', 'weight': 0.5}
{'vertexA': '02119-10', 'vertexB': '03019-10', 'weight': 1.0}
{'vertexA': '03019-10', 'vertexB': '03059-10', 'weight': 0.299999999999999716}
{'vertexA': '03059-10', 'vertexB': '03129-10', 'weight': 0.400000000000000213}
{'vertexA': '03129-10', 'vertexB': '03217-10', 'weight': 0.39999999999999986}
{'vertexA': '03217-10', 'vertexB': '05649-10', 'weight': 0.90000000000000021}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.299999999999999716}
{'vertexA': '14061-10', 'vertexB': '14071-10', 'weight': 0.5}
{'vertexA': '14071-10', 'vertexB': '14081-10', 'weight': 0.400000000000000213}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.39999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.80000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.30000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.39999999999999986}
{'vertexA': '14131-10', 'vertexB': '14151-10', 'weight': 0.5}
{'vertexA': '14151-10', 'vertexB': '14161-10', 'weight': 0.400000000000000213}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.39999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.30000000000000007}
El tiempo de ejecución es: 0.6280000000000174

```

Archivo: bus\_routes\_300.csv

\*\*\*\*\*

Seleccione una opción para continuar

>6

Estación destino (Ej: 15151-10): 15151-10

El camino es de longitud: 65

```
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.400000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.200000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.400000000000000036}
{'vertexA': '85049-10', 'vertexB': '85039-10', 'weight': 0.2999999999999998}
{'vertexA': '85039-10', 'vertexB': '85029-10', 'weight': 0.200000000000000018}
{'vertexA': '85029-10', 'vertexB': '85019-10', 'weight': 0.200000000000000018}
{'vertexA': '85019-10', 'vertexB': '94079-10', 'weight': 0.2999999999999998}
{'vertexA': '94079-10', 'vertexB': '94069-10', 'weight': 0.5999999999999996}
{'vertexA': '94069-10', 'vertexB': '94059-10', 'weight': 0.30000000000000007}
{'vertexA': '94059-10', 'vertexB': '94049-10', 'weight': 0.39999999999999947}
{'vertexA': '94049-10', 'vertexB': '94089-10', 'weight': 0.200000000000000018}
{'vertexA': '94089-10', 'vertexB': '94039-10', 'weight': 0.2999999999999998}
{'vertexA': '94039-10', 'vertexB': '94029-10', 'weight': 0.400000000000000036}
{'vertexA': '94029-10', 'vertexB': '94019-10', 'weight': 0.200000000000000018}
{'vertexA': '94019-10', 'vertexB': '93099-10', 'weight': 0.400000000000000036}
{'vertexA': '93099-10', 'vertexB': '93089-10', 'weight': 0.5}
{'vertexA': '93089-10', 'vertexB': '93079-10', 'weight': 0.29999999999999983}
{'vertexA': '93079-10', 'vertexB': '93069-10', 'weight': 0.30000000000000007}
{'vertexA': '93069-10', 'vertexB': '93059-10', 'weight': 0.400000000000000036}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999983}
{'vertexA': '92159-10', 'vertexB': '92149-10', 'weight': 0.30000000000000007}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999983}
{'vertexA': '92139-10', 'vertexB': '92129-10', 'weight': 0.5}
{'vertexA': '92129-10', 'vertexB': '92119-10', 'weight': 0.5}
{'vertexA': '92119-10', 'vertexB': '92109-10', 'weight': 0.200000000000000017}
{'vertexA': '92109-10', 'vertexB': '92099-10', 'weight': 0.29999999999999983}
{'vertexA': '92099-10', 'vertexB': '92161-10', 'weight': 0.5}
{'vertexA': '92161-10', 'vertexB': '82071-10', 'weight': 0.400000000000000036}
{'vertexA': '82071-10', 'vertexB': '81199-10', 'weight': 0.80000000000000007}
{'vertexA': '81199-10', 'vertexB': '81189-10', 'weight': 0.29999999999999983}
{'vertexA': '81189-10', 'vertexB': '81179-10', 'weight': 0.5}
{'vertexA': '81179-10', 'vertexB': '80141-10', 'weight': 0.600000000000000014}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80169-10', 'weight': 1.3999999999999986}
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.700000000000000028}
{'vertexA': '80159-10', 'vertexB': '02119-10', 'weight': 0.5}
{'vertexA': '02119-10', 'vertexB': '03019-10', 'weight': 1.0}
{'vertexA': '03019-10', 'vertexB': '03059-10', 'weight': 0.299999999999999716}
{'vertexA': '03059-10', 'vertexB': '03059-100', 'weight': 0}
{'vertexA': '03059-100', 'vertexB': '03129-100', 'weight': 0.39999999999999986}
{'vertexA': '03129-100', 'vertexB': '03217-100', 'weight': 0.400000000000000036}
{'vertexA': '03217-100', 'vertexB': '05649-100', 'weight': 0.90000000000000004}
{'vertexA': '05649-100', 'vertexB': '05649-10', 'weight': 0}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.299999999999999716}
{'vertexA': '14061-10', 'vertexB': '14061-100', 'weight': 0}
{'vertexA': '14061-100', 'vertexB': '14071-100', 'weight': 0.5}
{'vertexA': '14071-100', 'vertexB': '14081-100', 'weight': 0.39999999999999986}
{'vertexA': '14081-100', 'vertexB': '14081-10', 'weight': 0}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.39999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.80000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.30000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.39999999999999986}
{'vertexA': '14131-10', 'vertexB': '14131-100', 'weight': 0}
{'vertexA': '14131-100', 'vertexB': '14151-100', 'weight': 0.5}
{'vertexA': '14151-100', 'vertexB': '14161-100', 'weight': 0.39999999999999986}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.39999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.30000000000000007}
```

El tiempo de ejecución es: 0.6280000000000174

**Archivo:** bus\_routes\_1000.csv

\*\*\*\*\*

Seleccione una opción para continuar

>6

Estación destino (Ej: 15151-10): 15151-10

El camino es de longitud: 63

```
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.40000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.20000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.40000000000000036}
{'vertexA': '85049-10', 'vertexB': '85049-10e', 'weight': 0}
{'vertexA': '85049-10e', 'vertexB': '85039-10e', 'weight': 0.3}
{'vertexA': '85039-10e', 'vertexB': '85029-10e', 'weight': 0.2}
{'vertexA': '85029-10e', 'vertexB': '85019-10e', 'weight': 0.1999999999999996}
{'vertexA': '85019-10e', 'vertexB': '94079-10e', 'weight': 0.3000000000000004}
{'vertexA': '94079-10e', 'vertexB': '94069-10e', 'weight': 0.6000000000000001}
{'vertexA': '94069-10e', 'vertexB': '94059-10e', 'weight': 0.2999999999999998}
{'vertexA': '94059-10e', 'vertexB': '94049-10e', 'weight': 0.3999999999999999}
{'vertexA': '94049-10e', 'vertexB': '94089-10e', 'weight': 0.20000000000000018}
{'vertexA': '94089-10e', 'vertexB': '94039-10e', 'weight': 0.2999999999999998}
{'vertexA': '94039-10e', 'vertexB': '94029-10e', 'weight': 0.40000000000000036}
{'vertexA': '94029-10e', 'vertexB': '94019-10e', 'weight': 0.19999999999999973}
{'vertexA': '94019-10e', 'vertexB': '93099-10e', 'weight': 0.3999999999999999}
{'vertexA': '93099-10e', 'vertexB': '93089-10e', 'weight': 0.5}
{'vertexA': '93089-10e', 'vertexB': '93079-10e', 'weight': 0.2999999999999998}
{'vertexA': '93079-10e', 'vertexB': '93069-10e', 'weight': 0.30000000000000007}
{'vertexA': '93069-10e', 'vertexB': '93059-10e', 'weight': 0.39999999999999947}
{'vertexA': '93059-10e', 'vertexB': '93059-10', 'weight': 0}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999983}
{'vertexA': '92159-10', 'vertexB': '92159-10e', 'weight': 0}
{'vertexA': '92159-10e', 'vertexB': '92149-10e', 'weight': 0.30000000000000007}
{'vertexA': '92149-10e', 'vertexB': '92139-10e', 'weight': 0.2999999999999998}
{'vertexA': '92139-10e', 'vertexB': '92129-10e', 'weight': 0.5}
{'vertexA': '92129-10e', 'vertexB': '92119-10e', 'weight': 0.5}
{'vertexA': '92119-10e', 'vertexB': '92109-10e', 'weight': 0.20000000000000018}
{'vertexA': '92109-10e', 'vertexB': '92109-10', 'weight': 0}
{'vertexA': '92109-10', 'vertexB': '92099-10', 'weight': 0.29999999999999983}
{'vertexA': '92099-10', 'vertexB': '92099-10e', 'weight': 0}
{'vertexA': '92099-10e', 'vertexB': '80159-10e', 'weight': 5.8999999999999995}
{'vertexA': '80159-10e', 'vertexB': '02119-10e', 'weight': 0.5}
{'vertexA': '02119-10e', 'vertexB': '03019-10e', 'weight': 1.0}
{'vertexA': '03019-10e', 'vertexB': '03019-107', 'weight': 0}
{'vertexA': '03019-107', 'vertexB': '03059-107', 'weight': 0.29999999999999983}
{'vertexA': '03059-107', 'vertexB': '03059-100', 'weight': 0}
{'vertexA': '03059-100', 'vertexB': '03129-100', 'weight': 0.39999999999999986}
{'vertexA': '03129-100', 'vertexB': '03217-100', 'weight': 0.40000000000000036}
{'vertexA': '03217-100', 'vertexB': '05649-100', 'weight': 0.9000000000000004}
{'vertexA': '05649-100', 'vertexB': '05649-10', 'weight': 0}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.299999999999999716}
{'vertexA': '14061-10', 'vertexB': '14061-100', 'weight': 0}
{'vertexA': '14061-100', 'vertexB': '14071-100', 'weight': 0.5}
{'vertexA': '14071-100', 'vertexB': '14081-100', 'weight': 0.39999999999999986}
{'vertexA': '14081-100', 'vertexB': '14081-10', 'weight': 0}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.39999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.80000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.30000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.39999999999999986}
{'vertexA': '14131-10', 'vertexB': '14131-100', 'weight': 0}
{'vertexA': '14131-100', 'vertexB': '14151-100', 'weight': 0.5}
{'vertexA': '14151-100', 'vertexB': '14161-100', 'weight': 0.39999999999999986}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.39999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.30000000000000007}
```

El tiempo de ejecución es: 0.6950000000000012

**Archivo:** bus\_routes\_2000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 80
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.40000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.20000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.40000000000000036}
{'vertexA': '85049-10', 'vertexB': '85049-10e', 'weight': 0}
{'vertexA': '85049-10e', 'vertexB': '85039-10e', 'weight': 0.3}
{'vertexA': '85039-10e', 'vertexB': '85029-10e', 'weight': 0.2}
{'vertexA': '85029-10e', 'vertexB': '85019-10e', 'weight': 0.19999999999999996}
{'vertexA': '85019-10e', 'vertexB': '94079-10e', 'weight': 0.30000000000000004}
{'vertexA': '94079-10e', 'vertexB': '94069-10e', 'weight': 0.6000000000000001}
{'vertexA': '94069-10e', 'vertexB': '94059-10e', 'weight': 0.2999999999999998}
{'vertexA': '94059-10e', 'vertexB': '94049-10e', 'weight': 0.3999999999999999}
{'vertexA': '94049-10e', 'vertexB': '94089-10e', 'weight': 0.20000000000000018}
{'vertexA': '94089-10e', 'vertexB': '94039-10e', 'weight': 0.2999999999999998}
{'vertexA': '94039-10e', 'vertexB': '94029-10e', 'weight': 0.40000000000000036}
{'vertexA': '94029-10e', 'vertexB': '94019-10e', 'weight': 0.19999999999999973}
{'vertexA': '94019-10e', 'vertexB': '93099-10e', 'weight': 0.3999999999999999}
{'vertexA': '93099-10e', 'vertexB': '93089-10e', 'weight': 0.5}
{'vertexA': '93089-10e', 'vertexB': '93079-10e', 'weight': 0.2999999999999998}
{'vertexA': '93079-10e', 'vertexB': '93079-12', 'weight': 0}
{'vertexA': '93079-12', 'vertexB': '93069-12', 'weight': 0.29999999999999893}
{'vertexA': '93069-12', 'vertexB': '93069-10e', 'weight': 0}
{'vertexA': '93069-10e', 'vertexB': '93069-10', 'weight': 0}
{'vertexA': '93069-10', 'vertexB': '93059-10', 'weight': 0.40000000000000036}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999893}
{'vertexA': '92159-10', 'vertexB': '92159-10e', 'weight': 0}
{'vertexA': '92159-10e', 'vertexB': '92159-12', 'weight': 0}
{'vertexA': '92159-12', 'vertexB': '92149-12', 'weight': 0.29999999999999893}
{'vertexA': '92149-12', 'vertexB': '92149-10e', 'weight': 0}
{'vertexA': '92149-10e', 'vertexB': '92149-10', 'weight': 0}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999893}
{'vertexA': '92139-10', 'vertexB': '92139-10e', 'weight': 0}
{'vertexA': '92139-10e', 'vertexB': '92139-12', 'weight': 0}
{'vertexA': '92139-12', 'vertexB': '92129-12', 'weight': 0.5}
{'vertexA': '92129-12', 'vertexB': '92119-12', 'weight': 0.5}
{'vertexA': '92119-12', 'vertexB': '92109-12', 'weight': 0.1999999999999993}
{'vertexA': '92109-12', 'vertexB': '92109-10e', 'weight': 0}
{'vertexA': '92109-10e', 'vertexB': '92099-10e', 'weight': 0.2999999999999998}
{'vertexA': '92099-10e', 'vertexB': '92099-12', 'weight': 0}
{'vertexA': '92099-12', 'vertexB': '91079-12', 'weight': 0.40000000000000036}
{'vertexA': '91079-12', 'vertexB': '91069-12', 'weight': 0.29999999999999893}
{'vertexA': '91069-12', 'vertexB': '91059-12', 'weight': 0.60000000000000014}
{'vertexA': '91059-12', 'vertexB': '91049-12', 'weight': 0.1999999999999993}
{'vertexA': '91049-12', 'vertexB': '91091-12', 'weight': 0.3999999999999986}
{'vertexA': '91091-12', 'vertexB': '80271-12', 'weight': 0.30000000000000007}
{'vertexA': '80271-12', 'vertexB': '80141-12', 'weight': 0.400000000000000213}
{'vertexA': '80141-12', 'vertexB': '80141-11', 'weight': 0}
{'vertexA': '80141-11', 'vertexB': '80141-10', 'weight': 0}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80169-10', 'weight': 1.3999999999999986}
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.70000000000000028}
{'vertexA': '80159-10', 'vertexB': '02119-10', 'weight': 0.5}

```

```
{'vertexA': '80159-10', 'vertexB': '02119-10', 'weight': 0.5}
{'vertexA': '02119-10', 'vertexB': '03019-10', 'weight': 1.0}
{'vertexA': '03019-10', 'vertexB': '03059-10', 'weight': 0.29999999999999716}
{'vertexA': '03059-10', 'vertexB': '03059-100', 'weight': 0}
{'vertexA': '03059-100', 'vertexB': '03129-100', 'weight': 0.3999999999999986}
{'vertexA': '03129-100', 'vertexB': '03217-100', 'weight': 0.40000000000000036}
{'vertexA': '03217-100', 'vertexB': '05649-100', 'weight': 0.9000000000000004}
{'vertexA': '05649-100', 'vertexB': '05649-10', 'weight': 0}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.29999999999999716}
{'vertexA': '14061-10', 'vertexB': '14061-100', 'weight': 0}
{'vertexA': '14061-100', 'vertexB': '14071-100', 'weight': 0.5}
{'vertexA': '14071-100', 'vertexB': '14081-100', 'weight': 0.3999999999999986}
{'vertexA': '14081-100', 'vertexB': '14081-10', 'weight': 0}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.3999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.8000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.3000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.3999999999999986}
{'vertexA': '14131-10', 'vertexB': '14131-100', 'weight': 0}
{'vertexA': '14131-100', 'vertexB': '14151-100', 'weight': 0.5}
{'vertexA': '14151-100', 'vertexB': '14161-100', 'weight': 0.3999999999999986}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.3999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.3000000000000007}
El tiempo de ejecución es: 0.8300000000001084
```

**Archivo:** bus\_routes\_3000.csv



```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 81
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '96289-10', 'weight': 1.1999999999999997}
{'vertexA': '96289-10', 'vertexB': '96109-10', 'weight': 0.400000000000000036}
{'vertexA': '96109-10', 'vertexB': '85079-10', 'weight': 0.5999999999999996}
{'vertexA': '85079-10', 'vertexB': '85089-10', 'weight': 0.200000000000000018}
{'vertexA': '85089-10', 'vertexB': '85069-10', 'weight': 0.2999999999999998}
{'vertexA': '85069-10', 'vertexB': '85059-10', 'weight': 0.2999999999999998}
{'vertexA': '85059-10', 'vertexB': '85049-10', 'weight': 0.400000000000000036}
{'vertexA': '85049-10', 'vertexB': '85049-10e', 'weight': 0}
{'vertexA': '85049-10e', 'vertexB': '85039-10e', 'weight': 0.3}
{'vertexA': '85039-10e', 'vertexB': '85029-10e', 'weight': 0.2}
{'vertexA': '85029-10e', 'vertexB': '85019-10e', 'weight': 0.19999999999999996}
{'vertexA': '85019-10e', 'vertexB': '94079-10e', 'weight': 0.30000000000000004}
{'vertexA': '94079-10e', 'vertexB': '94069-10e', 'weight': 0.60000000000000001}
{'vertexA': '94069-10e', 'vertexB': '94059-10e', 'weight': 0.2999999999999998}
{'vertexA': '94059-10e', 'vertexB': '94049-10e', 'weight': 0.3999999999999999}
{'vertexA': '94049-10e', 'vertexB': '94039-10e', 'weight': 0.200000000000000018}
{'vertexA': '94039-10e', 'vertexB': '94029-10e', 'weight': 0.2999999999999998}
{'vertexA': '94029-10e', 'vertexB': '94019-10e', 'weight': 0.400000000000000036}
{'vertexA': '94019-10e', 'vertexB': '93099-10e', 'weight': 0.19999999999999973}
{'vertexA': '93099-10e', 'vertexB': '93089-10e', 'weight': 0.3999999999999999}
{'vertexA': '93089-10e', 'vertexB': '93079-10e', 'weight': 0.5}
{'vertexA': '93079-10e', 'vertexB': '93079-12', 'weight': 0.2999999999999998}
{'vertexA': '93079-12', 'vertexB': '93069-12', 'weight': 0.29999999999999983}
{'vertexA': '93069-12', 'vertexB': '93069-10e', 'weight': 0}
{'vertexA': '93069-10e', 'vertexB': '93069-10', 'weight': 0}
{'vertexA': '93069-10', 'vertexB': '93059-10', 'weight': 0.400000000000000036}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999983}
{'vertexA': '92159-10', 'vertexB': '92159-10e', 'weight': 0}
{'vertexA': '92159-10e', 'vertexB': '92159-12', 'weight': 0}
{'vertexA': '92159-12', 'vertexB': '92149-12', 'weight': 0.29999999999999983}
{'vertexA': '92149-12', 'vertexB': '92149-10e', 'weight': 0}
{'vertexA': '92149-10e', 'vertexB': '92149-10', 'weight': 0}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999983}
{'vertexA': '92139-10', 'vertexB': '92139-10e', 'weight': 0}
{'vertexA': '92139-10e', 'vertexB': '92139-12', 'weight': 0}
{'vertexA': '92139-12', 'vertexB': '92129-12', 'weight': 0.5}
{'vertexA': '92129-12', 'vertexB': '92119-12', 'weight': 0.5}
{'vertexA': '92119-12', 'vertexB': '92109-12', 'weight': 0.19999999999999993}
{'vertexA': '92109-12', 'vertexB': '92109-10e', 'weight': 0}
{'vertexA': '92109-10e', 'vertexB': '92099-10e', 'weight': 0.2999999999999998}
{'vertexA': '92099-10e', 'vertexB': '92099-12', 'weight': 0}
{'vertexA': '92099-12', 'vertexB': '91079-12', 'weight': 0.400000000000000036}
{'vertexA': '91079-12', 'vertexB': '91069-12', 'weight': 0.29999999999999983}
{'vertexA': '91069-12', 'vertexB': '91059-12', 'weight': 0.600000000000000014}
{'vertexA': '91059-12', 'vertexB': '91049-12', 'weight': 0.19999999999999993}
{'vertexA': '91049-12', 'vertexB': '91091-12', 'weight': 0.39999999999999986}
{'vertexA': '91091-12', 'vertexB': '80271-12', 'weight': 0.30000000000000007}
{'vertexA': '80271-12', 'vertexB': '80141-12', 'weight': 0.400000000000000213}
{'vertexA': '80141-12', 'vertexB': '80141-11', 'weight': 0}
{'vertexA': '80141-11', 'vertexB': '80141-10', 'weight': 0}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80169-10', 'weight': 1.3999999999999996}
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.70000000000000028}
{'vertexA': '80159-10', 'vertexB': '80159-10e', 'weight': 0}

```

```
{'vertexA': '80169-10', 'vertexB': '80159-10', 'weight': 0.7000000000000028}
{'vertexA': '80159-10', 'vertexB': '80159-10e', 'weight': 0}
{'vertexA': '80159-10e', 'vertexB': '02119-10e', 'weight': 0.5}
{'vertexA': '02119-10e', 'vertexB': '02119-128', 'weight': 0}
{'vertexA': '02119-128', 'vertexB': '03019-128', 'weight': 0.9999999999999982}
{'vertexA': '03019-128', 'vertexB': '03019-130', 'weight': 0}
{'vertexA': '03019-130', 'vertexB': '03019-131', 'weight': 0}
{'vertexA': '03019-131', 'vertexB': '03059-131', 'weight': 0.29999999999999893}
{'vertexA': '03059-131', 'vertexB': '03129-131', 'weight': 0.40000000000000036}
{'vertexA': '03129-131', 'vertexB': '03217-131', 'weight': 0.40000000000000036}
{'vertexA': '03217-131', 'vertexB': '05649-131', 'weight': 0.9000000000000004}
{'vertexA': '05649-131', 'vertexB': '14061-131', 'weight': 0.29999999999999893}
{'vertexA': '14061-131', 'vertexB': '14071-131', 'weight': 0.5}
{'vertexA': '14071-131', 'vertexB': '14081-131', 'weight': 0.40000000000000036}
{'vertexA': '14081-131', 'vertexB': '14101-131', 'weight': 0.40000000000000036}
{'vertexA': '14101-131', 'vertexB': '14141-131', 'weight': 0.8000000000000007}
{'vertexA': '14141-131', 'vertexB': '14121-131', 'weight': 0.29999999999999893}
{'vertexA': '14121-131', 'vertexB': '14131-131', 'weight': 0.40000000000000036}
{'vertexA': '14131-131', 'vertexB': '14131-124', 'weight': 0}
{'vertexA': '14131-124', 'vertexB': '14131-100', 'weight': 0}
{'vertexA': '14131-100', 'vertexB': '14151-100', 'weight': 0.5}
{'vertexA': '14151-100', 'vertexB': '14161-100', 'weight': 0.3999999999999986}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.3999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.3000000000000007}
El tiempo de ejecución es: 0.84000000000001739
```

**Archivo:** bus\_routes\_7000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 95
{'vertexA': '75009-10', 'vertexB': '76059-10', 'weight': 0.6}
{'vertexA': '76059-10', 'vertexB': '76069-10', 'weight': 0.5000000000000001}
{'vertexA': '76069-10', 'vertexB': '76069-15', 'weight': 0}
{'vertexA': '76069-15', 'vertexB': '76069-15A', 'weight': 0}
{'vertexA': '76069-15A', 'vertexB': '76069-168', 'weight': 0}
{'vertexA': '76069-168', 'vertexB': '75059-168', 'weight': 0.6999999999999993}
{'vertexA': '75059-168', 'vertexB': '75069-168', 'weight': 0.70000000000000028}
{'vertexA': '75069-168', 'vertexB': '75349-168', 'weight': 0.3999999999999986}
{'vertexA': '75349-168', 'vertexB': '84209-168', 'weight': 0.1999999999999993}
{'vertexA': '84209-168', 'vertexB': '84529-168', 'weight': 0.5}
{'vertexA': '84529-168', 'vertexB': '84219-168', 'weight': 0.30000000000000007}
{'vertexA': '84219-168', 'vertexB': '84219-14e', 'weight': 0}
{'vertexA': '84219-14e', 'vertexB': '84219-14', 'weight': 0}
{'vertexA': '84219-14', 'vertexB': '84351-14', 'weight': 0.29999999999999716}
{'vertexA': '84351-14', 'vertexB': '84351-155', 'weight': 0}
{'vertexA': '84351-155', 'vertexB': '84351-168', 'weight': 0}
{'vertexA': '84351-168', 'vertexB': '84009-168', 'weight': 0.6999999999999993}
{'vertexA': '84009-168', 'vertexB': '84009-16', 'weight': 0}
{'vertexA': '84009-16', 'vertexB': '84139-16', 'weight': 0.7}
{'vertexA': '84139-16', 'vertexB': '84129-16', 'weight': 0.6000000000000001}
{'vertexA': '84129-16', 'vertexB': '84129-137', 'weight': 0}
{'vertexA': '84129-137', 'vertexB': '84151-137', 'weight': 0.5}
{'vertexA': '84151-137', 'vertexB': '84151-12e', 'weight': 0}
{'vertexA': '84151-12e', 'vertexB': '84151-12', 'weight': 0}
{'vertexA': '84151-12', 'vertexB': '93099-12', 'weight': 0.5999999999999996}
{'vertexA': '93099-12', 'vertexB': '93099-10e', 'weight': 0}
{'vertexA': '93099-10e', 'vertexB': '93099-10', 'weight': 0}
{'vertexA': '93099-10', 'vertexB': '93089-10', 'weight': 0.5}
{'vertexA': '93089-10', 'vertexB': '93079-10', 'weight': 0.29999999999999893}
{'vertexA': '93079-10', 'vertexB': '93079-10e', 'weight': 0}
{'vertexA': '93079-10e', 'vertexB': '93079-12', 'weight': 0}
{'vertexA': '93079-12', 'vertexB': '93069-12', 'weight': 0.29999999999999893}
{'vertexA': '93069-12', 'vertexB': '93069-10e', 'weight': 0}
{'vertexA': '93069-10e', 'vertexB': '93059-10e', 'weight': 0.39999999999999947}
{'vertexA': '93059-10e', 'vertexB': '93059-10', 'weight': 0}
{'vertexA': '93059-10', 'vertexB': '92159-10', 'weight': 0.29999999999999893}
{'vertexA': '92159-10', 'vertexB': '92159-10e', 'weight': 0}
{'vertexA': '92159-10e', 'vertexB': '92159-12', 'weight': 0}
{'vertexA': '92159-12', 'vertexB': '92149-12', 'weight': 0.29999999999999893}
{'vertexA': '92149-12', 'vertexB': '92149-10e', 'weight': 0}
{'vertexA': '92149-10e', 'vertexB': '92149-10', 'weight': 0}
{'vertexA': '92149-10', 'vertexB': '92139-10', 'weight': 0.29999999999999893}
{'vertexA': '92139-10', 'vertexB': '92139-10e', 'weight': 0}
{'vertexA': '92139-10e', 'vertexB': '92139-12', 'weight': 0}
{'vertexA': '92139-12', 'vertexB': '92129-12', 'weight': 0.5}
{'vertexA': '92129-12', 'vertexB': '92119-12', 'weight': 0.5}
{'vertexA': '92119-12', 'vertexB': '92109-12', 'weight': 0.1999999999999993}
{'vertexA': '92109-12', 'vertexB': '92109-10e', 'weight': 0}
{'vertexA': '92109-10e', 'vertexB': '92099-10e', 'weight': 0.2999999999999998}
{'vertexA': '92099-10e', 'vertexB': '92099-12', 'weight': 0}
{'vertexA': '92099-12', 'vertexB': '91079-12', 'weight': 0.40000000000000036}
{'vertexA': '91079-12', 'vertexB': '91069-12', 'weight': 0.29999999999999893}
{'vertexA': '91069-12', 'vertexB': '91059-12', 'weight': 0.60000000000000014}
{'vertexA': '91059-12', 'vertexB': '91049-12', 'weight': 0.1999999999999993}
{'vertexA': '91049-12', 'vertexB': '91091-12', 'weight': 0.3999999999999986}
{'vertexA': '91091-12', 'vertexB': '91091-14', 'weight': 0}
{'vertexA': '91091-14', 'vertexB': '91091-14A', 'weight': 0}

```

```

{'vertexA': '91079-12', 'vertexB': '91069-12', 'weight': 0.2999999999999893}
{'vertexA': '91069-12', 'vertexB': '91059-12', 'weight': 0.6000000000000014}
{'vertexA': '91059-12', 'vertexB': '91049-12', 'weight': 0.1999999999999993}
{'vertexA': '91049-12', 'vertexB': '91091-12', 'weight': 0.3999999999999986}
{'vertexA': '91091-12', 'vertexB': '91091-14', 'weight': 0}
{'vertexA': '91091-14', 'vertexB': '91091-14A', 'weight': 0}
{'vertexA': '91091-14A', 'vertexB': '80271-14A', 'weight': 0.2999999999999893}
{'vertexA': '80271-14A', 'vertexB': '80271-158', 'weight': 0}
{'vertexA': '80271-158', 'vertexB': '80141-158', 'weight': 0.3999999999999986}
{'vertexA': '80141-158', 'vertexB': '80141-14A', 'weight': 0}
{'vertexA': '80141-14A', 'vertexB': '80141-14', 'weight': 0}
{'vertexA': '80141-14', 'vertexB': '80141-12', 'weight': 0}
{'vertexA': '80141-12', 'vertexB': '80141-11', 'weight': 0}
{'vertexA': '80141-11', 'vertexB': '80141-10', 'weight': 0}
{'vertexA': '80141-10', 'vertexB': '80219-10', 'weight': 0.6999999999999993}
{'vertexA': '80219-10', 'vertexB': '80219-14', 'weight': 0}
{'vertexA': '80219-14', 'vertexB': '80169-14', 'weight': 1.3999999999999986}
{'vertexA': '80169-14', 'vertexB': '80159-14', 'weight': 0.6999999999999993}
{'vertexA': '80159-14', 'vertexB': '80159-10e', 'weight': 0}
{'vertexA': '80159-10e', 'vertexB': '02119-10e', 'weight': 0.5}
{'vertexA': '02119-10e', 'vertexB': '03019-10e', 'weight': 1.0}
{'vertexA': '03019-10e', 'vertexB': '03019-107', 'weight': 0}
{'vertexA': '03019-107', 'vertexB': '03059-107', 'weight': 0.2999999999999893}
{'vertexA': '03059-107', 'vertexB': '03059-100', 'weight': 0}
{'vertexA': '03059-100', 'vertexB': '03129-100', 'weight': 0.3999999999999986}
{'vertexA': '03129-100', 'vertexB': '03217-100', 'weight': 0.4000000000000036}
{'vertexA': '03217-100', 'vertexB': '05649-100', 'weight': 0.9000000000000004}
{'vertexA': '05649-100', 'vertexB': '05649-10', 'weight': 0}
{'vertexA': '05649-10', 'vertexB': '14061-10', 'weight': 0.2999999999999716}
{'vertexA': '14061-10', 'vertexB': '14061-100', 'weight': 0}
{'vertexA': '14061-100', 'vertexB': '14071-100', 'weight': 0.5}
{'vertexA': '14071-100', 'vertexB': '14081-100', 'weight': 0.3999999999999986}
{'vertexA': '14081-100', 'vertexB': '14081-10', 'weight': 0}
{'vertexA': '14081-10', 'vertexB': '14101-10', 'weight': 0.3999999999999986}
{'vertexA': '14101-10', 'vertexB': '14141-10', 'weight': 0.8000000000000007}
{'vertexA': '14141-10', 'vertexB': '14121-10', 'weight': 0.3000000000000007}
{'vertexA': '14121-10', 'vertexB': '14131-10', 'weight': 0.3999999999999986}
{'vertexA': '14131-10', 'vertexB': '14131-100', 'weight': 0}
{'vertexA': '14131-100', 'vertexB': '14151-100', 'weight': 0.5}
{'vertexA': '14151-100', 'vertexB': '14161-100', 'weight': 0.3999999999999986}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.3999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.3000000000000007}
El tiempo de ejecución es: 0.9709999999998331

```

**Archivo:** bus\_routes\_10000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 112
{'vertexA': '75009-10', 'vertexB': '75009-127', 'weight': 0}
{'vertexA': '75009-127', 'vertexB': '75009-127A', 'weight': 0}
{'vertexA': '75009-127A', 'vertexB': '75009-19', 'weight': 0}
{'vertexA': '75009-19', 'vertexB': '75009-20', 'weight': 0}
{'vertexA': '75009-20', 'vertexB': '76059-20', 'weight': 0.6}
{'vertexA': '76059-20', 'vertexB': '76069-20', 'weight': 0.5000000000000001}
{'vertexA': '76069-20', 'vertexB': '76069-168', 'weight': 0}
{'vertexA': '76069-168', 'vertexB': '75059-168', 'weight': 0.6999999999999993}
{'vertexA': '75059-168', 'vertexB': '75059-18', 'weight': 0}
{'vertexA': '75059-18', 'vertexB': '75069-18', 'weight': 0.7000000000000002}
{'vertexA': '75069-18', 'vertexB': '75069-168', 'weight': 0}
{'vertexA': '75069-168', 'vertexB': '75349-168', 'weight': 0.3999999999999986}
{'vertexA': '75349-168', 'vertexB': '75349-18', 'weight': 0}
{'vertexA': '75349-18', 'vertexB': '75349-21', 'weight': 0}
{'vertexA': '75349-21', 'vertexB': '84609-21', 'weight': 0.29999999999999893}
{'vertexA': '84609-21', 'vertexB': '84639-21', 'weight': 0.30000000000000007}
{'vertexA': '84639-21', 'vertexB': '84619-21', 'weight': 0.1999999999999993}
{'vertexA': '84619-21', 'vertexB': '84629-21', 'weight': 0.30000000000000007}
{'vertexA': '84629-21', 'vertexB': '84479-21', 'weight': 0.29999999999999893}
{'vertexA': '84479-21', 'vertexB': '84469-21', 'weight': 0.40000000000000036}
{'vertexA': '84469-21', 'vertexB': '84481-21', 'weight': 0.5999999999999996}
{'vertexA': '84481-21', 'vertexB': '84491-21', 'weight': 0.5}
{'vertexA': '84491-21', 'vertexB': '72079-21', 'weight': 0.40000000000000036}
{'vertexA': '72079-21', 'vertexB': '72069-21', 'weight': 0.30000000000000007}
{'vertexA': '72069-21', 'vertexB': '72029-21', 'weight': 0.40000000000000036}
{'vertexA': '72029-21', 'vertexB': '72019-21', 'weight': 0.3999999999999986}
{'vertexA': '72019-21', 'vertexB': '72019-15A', 'weight': 0}
{'vertexA': '72019-15A', 'vertexB': '83119-15A', 'weight': 0.6999999999999993}
{'vertexA': '83119-15A', 'vertexB': '83119-21', 'weight': 0}
{'vertexA': '83119-21', 'vertexB': '83109-21', 'weight': 0.1999999999999993}
{'vertexA': '83109-21', 'vertexB': '82069-21', 'weight': 0.70000000000000011}
{'vertexA': '82069-21', 'vertexB': '82069-2', 'weight': 0}
{'vertexA': '82069-2', 'vertexB': '82049-2', 'weight': 0.5}
{'vertexA': '82049-2', 'vertexB': '82049-155', 'weight': 0}
{'vertexA': '82049-155', 'vertexB': '82029-155', 'weight': 0.3999999999999986}
{'vertexA': '82029-155', 'vertexB': '82029-2', 'weight': 0}
{'vertexA': '82029-2', 'vertexB': '82109-2', 'weight': 0.29999999999999716}
{'vertexA': '82109-2', 'vertexB': '82109-13', 'weight': 0}
{'vertexA': '82109-13', 'vertexB': '81069-13', 'weight': 0.30000000000000007}
{'vertexA': '81069-13', 'vertexB': '81049-13', 'weight': 0.5}
{'vertexA': '81049-13', 'vertexB': '81029-13', 'weight': 0.29999999999999893}
{'vertexA': '81029-13', 'vertexB': '80089-13', 'weight': 0.5}
{'vertexA': '80089-13', 'vertexB': '80089-100', 'weight': 0}
{'vertexA': '80089-100', 'vertexB': '80069-100', 'weight': 0.39999999999999947}
{'vertexA': '80069-100', 'vertexB': '80069-13', 'weight': 0}
{'vertexA': '80069-13', 'vertexB': '80049-13', 'weight': 0.5}
{'vertexA': '80049-13', 'vertexB': '80039-13', 'weight': 0.1999999999999993}
{'vertexA': '80039-13', 'vertexB': '80039-197', 'weight': 0}
{'vertexA': '80039-197', 'vertexB': '80039-2', 'weight': 0}
{'vertexA': '80039-2', 'vertexB': '80029-2', 'weight': 0.5}
{'vertexA': '80029-2', 'vertexB': '01319-2', 'weight': 0.5}
{'vertexA': '01319-2', 'vertexB': '01219-2', 'weight': 0.3999999999999986}
{'vertexA': '01219-2', 'vertexB': '01129-2', 'weight': 0.30000000000000007}

```

```
{'vertexA': '80049-13', 'vertexB': '80039-13', 'weight': 0.1999999999999993}
{'vertexA': '80039-13', 'vertexB': '80039-197', 'weight': 0}
{'vertexA': '80039-197', 'vertexB': '80039-2', 'weight': 0}
{'vertexA': '80039-2', 'vertexB': '80029-2', 'weight': 0.5}
{'vertexA': '80029-2', 'vertexB': '01319-2', 'weight': 0.5}
{'vertexA': '01319-2', 'vertexB': '01219-2', 'weight': 0.3999999999999986}
{'vertexA': '01219-2', 'vertexB': '01129-2', 'weight': 0.3000000000000007}
{'vertexA': '01129-2', 'vertexB': '01059-2', 'weight': 0.3999999999999986}
{'vertexA': '01059-2', 'vertexB': '01059-133', 'weight': 0}
{'vertexA': '01059-133', 'vertexB': '01119-133', 'weight': 0.1999999999999993}
{'vertexA': '01119-133', 'vertexB': '01119-130', 'weight': 0}
{'vertexA': '01119-130', 'vertexB': '01119-12e', 'weight': 0}
{'vertexA': '01119-12e', 'vertexB': '01019-12e', 'weight': 0.3999999999999986}
{'vertexA': '01019-12e', 'vertexB': '01019-130', 'weight': 0}
{'vertexA': '01019-130', 'vertexB': '01019-133', 'weight': 0}
{'vertexA': '01019-133', 'vertexB': '01019-2', 'weight': 0}
{'vertexA': '01019-2', 'vertexB': '04159-2', 'weight': 0.3000000000000007}
{'vertexA': '04159-2', 'vertexB': '04149-2', 'weight': 0.29999999999999716}
{'vertexA': '04149-2', 'vertexB': '04149-190', 'weight': 0}
{'vertexA': '04149-190', 'vertexB': '04149-147A', 'weight': 0}
{'vertexA': '04149-147A', 'vertexB': '04149-147', 'weight': 0}
{'vertexA': '04149-147', 'vertexB': '04149-12e', 'weight': 0}
{'vertexA': '04149-12e', 'vertexB': '04149-12', 'weight': 0}
{'vertexA': '04149-12', 'vertexB': '04229-12', 'weight': 0.29999999999999716}
{'vertexA': '04229-12', 'vertexB': '04229-12e', 'weight': 0}
{'vertexA': '04229-12e', 'vertexB': '04239-12e', 'weight': 0.1999999999999993}
{'vertexA': '04239-12e', 'vertexB': '04239-12', 'weight': 0}
{'vertexA': '04239-12', 'vertexB': '05059-12', 'weight': 0.29999999999999716}
{'vertexA': '05059-12', 'vertexB': '05059-12e', 'weight': 0}
{'vertexA': '05059-12e', 'vertexB': '05059-147', 'weight': 0}
{'vertexA': '05059-147', 'vertexB': '05059-147A', 'weight': 0}
{'vertexA': '05059-147A', 'vertexB': '05059-147e', 'weight': 0}
{'vertexA': '05059-147e', 'vertexB': '05049-147e', 'weight': 0.2}
{'vertexA': '05049-147e', 'vertexB': '05039-147e', 'weight': 0.4}
{'vertexA': '05039-147e', 'vertexB': '05039-147A', 'weight': 0}
{'vertexA': '05039-147A', 'vertexB': '05039-147', 'weight': 0}
{'vertexA': '05039-147', 'vertexB': '05019-147', 'weight': 0.3000000000000007}
{'vertexA': '05019-147', 'vertexB': '05069-147', 'weight': 0.09999999999999787}
{'vertexA': '05069-147', 'vertexB': '05069-147e', 'weight': 0}
{'vertexA': '05069-147e', 'vertexB': '10011-147e', 'weight': 0.3999999999999999}
{'vertexA': '10011-147e', 'vertexB': '10011-174', 'weight': 0}
{'vertexA': '10011-174', 'vertexB': '10041-174', 'weight': 0.29999999999999716}
{'vertexA': '10041-174', 'vertexB': '10041-190', 'weight': 0}
{'vertexA': '10041-190', 'vertexB': '10041-196', 'weight': 0}
{'vertexA': '10041-196', 'vertexB': '10051-196', 'weight': 0.3999999999999986}
{'vertexA': '10051-196', 'vertexB': '10051-197', 'weight': 0}
{'vertexA': '10051-197', 'vertexB': '10061-197', 'weight': 0.3000000000000007}
{'vertexA': '10061-197', 'vertexB': '10071-197', 'weight': 0.3999999999999986}
{'vertexA': '10071-197', 'vertexB': '10501-197', 'weight': 0.1999999999999993}
{'vertexA': '10501-197', 'vertexB': '10501-196', 'weight': 0}
{'vertexA': '10501-196', 'vertexB': '10081-196', 'weight': 0.29999999999999716}
{'vertexA': '10081-196', 'vertexB': '10081-175', 'weight': 0}
{'vertexA': '10081-175', 'vertexB': '10339-175', 'weight': 0.5}
{'vertexA': '10339-175', 'vertexB': '10339-145A', 'weight': 0}
{'vertexA': '10339-145A', 'vertexB': '10339-145', 'weight': 0}
{'vertexA': '10339-145', 'vertexB': '14249-145', 'weight': 0.20000000000000018}
{'vertexA': '14249-145', 'vertexB': '14059-145', 'weight': 0.5}
{'vertexA': '14059-145', 'vertexB': '14059-131', 'weight': 0}
{'vertexA': '14059-131', 'vertexB': '14049-131', 'weight': 0.39999999999999947}
{'vertexA': '14049-131', 'vertexB': '14049-120', 'weight': 0}
{'vertexA': '14049-120', 'vertexB': '14161-120', 'weight': 0.5}
{'vertexA': '14161-120', 'vertexB': '14161-100', 'weight': 0}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.3999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.3000000000000007}
El tiempo de ejecución es: 1.1109999999980857
```

**Archivo:** bus\_routes\_14000.csv

```

*****
Bienvenido
1- Inicializar Analizador
2- Cargar información de buses de singapur
3- Calcular componentes conectados
4- Establecer estación base:
5- Hay camino entre estacion base y estación:
6- Ruta de costo mínimo desde la estación base y estación:
7- Estación que sirve a mas rutas:
0- Salir
*****
Seleccione una opción para continuar
>6
Estación destino (Ej: 15151-10): 15151-10
El camino es de longitud: 136
{'vertexA': '75009-10', 'vertexB': '75009-127', 'weight': 0}
{'vertexA': '75009-127', 'vertexB': '75009-127A', 'weight': 0}
{'vertexA': '75009-127A', 'vertexB': '75009-19', 'weight': 0}
{'vertexA': '75009-19', 'vertexB': '75009-20', 'weight': 0}
{'vertexA': '75009-20', 'vertexB': '75009-22', 'weight': 0}
{'vertexA': '75009-22', 'vertexB': '76059-22', 'weight': 0.6}
{'vertexA': '76059-22', 'vertexB': '76111-22', 'weight': 0.20000000000000007}
{'vertexA': '76111-22', 'vertexB': '76121-22', 'weight': 0.30000000000000004}
{'vertexA': '76121-22', 'vertexB': '76281-22', 'weight': 0.29999999999999998}
{'vertexA': '76281-22', 'vertexB': '75069-22', 'weight': 0.80000000000000003}
{'vertexA': '75069-22', 'vertexB': '75069-21', 'weight': 0}
{'vertexA': '75069-21', 'vertexB': '75069-18', 'weight': 0}
{'vertexA': '75069-18', 'vertexB': '75069-168', 'weight': 0}
{'vertexA': '75069-168', 'vertexB': '75349-168', 'weight': 0.39999999999999986}
{'vertexA': '75349-168', 'vertexB': '84209-168', 'weight': 0.19999999999999993}
{'vertexA': '84209-168', 'vertexB': '84209-18', 'weight': 0}
{'vertexA': '84209-18', 'vertexB': '84209-228', 'weight': 0}
{'vertexA': '84209-228', 'vertexB': '84209-28', 'weight': 0}
{'vertexA': '84209-28', 'vertexB': '84529-28', 'weight': 0.49999999999999991}
{'vertexA': '84529-28', 'vertexB': '84529-228', 'weight': 0}
{'vertexA': '84529-228', 'vertexB': '84219-228', 'weight': 0.29999999999999983}
{'vertexA': '84219-228', 'vertexB': '84219-222', 'weight': 0}
{'vertexA': '84219-222', 'vertexB': '84219-168', 'weight': 0}
{'vertexA': '84219-168', 'vertexB': '84219-14e', 'weight': 0}
{'vertexA': '84219-14e', 'vertexB': '84219-14', 'weight': 0}
{'vertexA': '84219-14', 'vertexB': '84351-14', 'weight': 0.299999999999999716}
{'vertexA': '84351-14', 'vertexB': '84351-155', 'weight': 0}
{'vertexA': '84351-155', 'vertexB': '84351-168', 'weight': 0}
{'vertexA': '84351-168', 'vertexB': '84351-222', 'weight': 0}
{'vertexA': '84351-222', 'vertexB': '84351-228', 'weight': 0}
{'vertexA': '84351-228', 'vertexB': '84009-228', 'weight': 0.69999999999999993}
{'vertexA': '84009-228', 'vertexB': '84009-229', 'weight': 0}
{'vertexA': '84009-229', 'vertexB': '84009-26', 'weight': 0}
{'vertexA': '84009-26', 'vertexB': '84009-30', 'weight': 0}
{'vertexA': '84009-30', 'vertexB': '84029-30', 'weight': 0.7}
{'vertexA': '84029-30', 'vertexB': '84029-28', 'weight': 0}
{'vertexA': '84029-28', 'vertexB': '84019-28', 'weight': 0.29999999999999983}
{'vertexA': '84019-28', 'vertexB': '84019-25', 'weight': 0}
{'vertexA': '84019-25', 'vertexB': '84019-24', 'weight': 0}
{'vertexA': '84019-24', 'vertexB': '83099-24', 'weight': 0.5}
{'vertexA': '83099-24', 'vertexB': '83079-24', 'weight': 0.39999999999999986}
{'vertexA': '83079-24', 'vertexB': '83059-24', 'weight': 0.39999999999999986}
{'vertexA': '83059-24', 'vertexB': '83059-25', 'weight': 0}
{'vertexA': '83059-25', 'vertexB': '83059-26', 'weight': 0}
{'vertexA': '83059-26', 'vertexB': '83059-28', 'weight': 0}
{'vertexA': '83059-28', 'vertexB': '83049-28', 'weight': 0.19999999999999993}
{'vertexA': '83049-28', 'vertexB': '83049-26', 'weight': 0}
{'vertexA': '83049-26', 'vertexB': '83049-25', 'weight': 0}

```



```

{'vertexA': '80049-32', 'vertexB': '80049-33', 'weight': 0}
{'vertexA': '80049-33', 'vertexB': '80039-33', 'weight': 0.1999999999999993}
{'vertexA': '80039-33', 'vertexB': '80039-32', 'weight': 0}
{'vertexA': '80039-32', 'vertexB': '80039-2', 'weight': 0}
{'vertexA': '80039-2', 'vertexB': '80029-2', 'weight': 0.5}
{'vertexA': '80029-2', 'vertexB': '01319-2', 'weight': 0.5}
{'vertexA': '01319-2', 'vertexB': '01219-2', 'weight': 0.3999999999999986}
{'vertexA': '01219-2', 'vertexB': '01219-33', 'weight': 0}
{'vertexA': '01219-33', 'vertexB': '01129-33', 'weight': 0.29999999999999893}
{'vertexA': '01129-33', 'vertexB': '01129-2', 'weight': 0}
{'vertexA': '01129-2', 'vertexB': '01059-2', 'weight': 0.3999999999999986}
{'vertexA': '01059-2', 'vertexB': '01119-2', 'weight': 0.1999999999999993}
{'vertexA': '01119-2', 'vertexB': '01119-33', 'weight': 0}
{'vertexA': '01119-33', 'vertexB': '01019-33', 'weight': 0.3999999999999986}
{'vertexA': '01019-33', 'vertexB': '01019-2', 'weight': 0}
{'vertexA': '01019-2', 'vertexB': '04159-2', 'weight': 0.30000000000000007}
{'vertexA': '04159-2', 'vertexB': '04149-2', 'weight': 0.29999999999999716}
{'vertexA': '04149-2', 'vertexB': '04149-190', 'weight': 0}
{'vertexA': '04149-190', 'vertexB': '04149-147A', 'weight': 0}
{'vertexA': '04149-147A', 'vertexB': '04149-147', 'weight': 0}
{'vertexA': '04149-147', 'vertexB': '04149-12e', 'weight': 0}
{'vertexA': '04149-12e', 'vertexB': '04149-12', 'weight': 0}
{'vertexA': '04149-12', 'vertexB': '04229-12', 'weight': 0.29999999999999716}
{'vertexA': '04229-12', 'vertexB': '04229-12e', 'weight': 0}
{'vertexA': '04229-12e', 'vertexB': '04239-12e', 'weight': 0.1999999999999993}
{'vertexA': '04239-12e', 'vertexB': '04239-12', 'weight': 0}
{'vertexA': '04239-12', 'vertexB': '05059-12', 'weight': 0.29999999999999716}
{'vertexA': '05059-12', 'vertexB': '05059-12e', 'weight': 0}
{'vertexA': '05059-12e', 'vertexB': '05059-147', 'weight': 0}
{'vertexA': '05059-147', 'vertexB': '05059-147A', 'weight': 0}
{'vertexA': '05059-147A', 'vertexB': '05059-147e', 'weight': 0}
{'vertexA': '05059-147e', 'vertexB': '05049-147e', 'weight': 0.2}
{'vertexA': '05049-147e', 'vertexB': '05039-147e', 'weight': 0.4}
{'vertexA': '05039-147e', 'vertexB': '05039-147A', 'weight': 0}
{'vertexA': '05039-147A', 'vertexB': '05039-147', 'weight': 0}
{'vertexA': '05039-147', 'vertexB': '05019-147', 'weight': 0.30000000000000007}
{'vertexA': '05019-147', 'vertexB': '05069-147', 'weight': 0.09999999999999787}
{'vertexA': '05069-147', 'vertexB': '05069-147e', 'weight': 0}
{'vertexA': '05069-147e', 'vertexB': '10011-147e', 'weight': 0.3999999999999999}
{'vertexA': '10011-147e', 'vertexB': '10011-174', 'weight': 0}
{'vertexA': '10011-174', 'vertexB': '10041-174', 'weight': 0.29999999999999716}
{'vertexA': '10041-174', 'vertexB': '10041-190', 'weight': 0}
{'vertexA': '10041-190', 'vertexB': '10041-196', 'weight': 0}
{'vertexA': '10041-196', 'vertexB': '10051-196', 'weight': 0.3999999999999986}
{'vertexA': '10051-196', 'vertexB': '10051-197', 'weight': 0}
{'vertexA': '10051-197', 'vertexB': '10061-197', 'weight': 0.30000000000000007}
{'vertexA': '10061-197', 'vertexB': '10071-197', 'weight': 0.3999999999999986}
{'vertexA': '10071-197', 'vertexB': '10501-197', 'weight': 0.1999999999999993}
{'vertexA': '10501-197', 'vertexB': '10501-196', 'weight': 0}
{'vertexA': '10501-196', 'vertexB': '10081-196', 'weight': 0.29999999999999716}
{'vertexA': '10081-196', 'vertexB': '10081-175', 'weight': 0}
{'vertexA': '10081-175', 'vertexB': '10339-175', 'weight': 0.5}
{'vertexA': '10339-175', 'vertexB': '10339-145A', 'weight': 0}
{'vertexA': '10339-145A', 'vertexB': '10339-145', 'weight': 0}
{'vertexA': '10339-145', 'vertexB': '14249-145', 'weight': 0.20000000000000018}
{'vertexA': '14249-145', 'vertexB': '14059-145', 'weight': 0.5}
{'vertexA': '14059-145', 'vertexB': '14059-131', 'weight': 0}
{'vertexA': '14059-131', 'vertexB': '14049-131', 'weight': 0.39999999999999947}
{'vertexA': '14049-131', 'vertexB': '14049-120', 'weight': 0}
{'vertexA': '14049-120', 'vertexB': '14161-120', 'weight': 0.5}
{'vertexA': '14161-120', 'vertexB': '14161-100', 'weight': 0}
{'vertexA': '14161-100', 'vertexB': '14161-10', 'weight': 0}
{'vertexA': '14161-10', 'vertexB': '14171-10', 'weight': 0.3999999999999986}
{'vertexA': '14171-10', 'vertexB': '15141-10', 'weight': 0.6999999999999993}
{'vertexA': '15141-10', 'vertexB': '15151-10', 'weight': 0.30000000000000007}

```

El tiempo de ejecución es: 1.3809999999949696

f) **¿Cuál es el tamaño inicial del grafo?**

El tamaño inicial del grafo es **14000**.

g) **¿Cuál es la Estructura de datos utilizada?**

Se utiliza un TAD grafo, que permite los calculos necesarios para las conexiones entre las rutas. Además, tambien se implementa una tabla de hash donde se almacenan los vértices del grafo.

h) **¿Cuál es la función de comparación utilizada?**

Se utiliza la función `compareStopIds`, la cual compara dos IDs.

```
270
271
272     def compareStopIds(stop, keyvaluestop):
273         """
274         Compara dos estaciones
275         """
276         stopcode = keyvaluestop['key']
277         if (stop == stopcode):
278             return 0
279         elif (stop > stopcode):
280             return 1
281         else:
282             return -1
283
```