Informe Proyecto Final

"A - Zlatan Galactic Summit"

Miguel Angel Padilla

Entrega: Noviembre 12, 2025

Universidad Pontificia Javeriana Cali

Árboles y Grafos

Profesor: Carlos Ramírez

Índice

1. Descripción general del problema

El problema exige encontrar la ciudad optima y el mínimo costo total de llegar a esa ciudad por todos los diferentes lideres del imperio, Siendo una ciudad optima aquella donde deben reunirse los lideres de las civilizaciones aliadas de manera que el costo total de energía de todos los viajes sea mínimo y, ademas, todos los lideres puedan llegar dentro del limite máximo de movimientos permitidos.

Los sectores están distribuidos en una cuadrícula cuadrada llamada El Imperio Galáctico de Zlatan. Cada sector (coordenada de la cuadricula) tiene un **coste de energía** asociado que representa lo que cuesta llegar a esa ciudad.

Los líderes, cada uno ubicado en una Ciudad distinta, pueden desplazarse en las cuatro direcciones cardinales (norte, sur, este, oeste). Cada vez que un líder llega a una Ciudad, paga la energía correspondiente a esa Ciudad; además, por motivos de seguridad y logística, **ningún líder puede desplazarse más de los movimientos máximos propuestos inicialmente** para llegar al punto de encuentro elegido.

El coste energético del sector que se elija como sede lo asume Zlatan, así que **ese coste no entra en la suma que queremos minimizar** (los líderes no pagan la energía del sector anfitrión al llegar).

Si no hay ninguna casilla a la que todos puedan llegar cumpliendo con el límite máximo de movimientos propuestos, la solución es "imposible".

2. Especificación del problema

2.1. Entrada

Recibe tres números N, F y T, donde

- $N \in \mathbb{N}$ es el numero de filas y de columnas de la cuadricula.
- $F \in \mathbb{N}$ es el número de líderes.
- $T \in \mathbb{N}$ es el número máximo de movimientos permitidos por líder.

Sea un conjunto de sectores representado por un grafo no dirigido $G = (V, E)$, donde los vértices V corresponden a las ciudades del Imperio y las aristas E representan las conexiones entre sectores adyacentes (norte, sur, este y oeste).

Este grafo se modela mediante una matriz de costos de energía:

$$M[0..N)[0..N) = \begin{bmatrix} a_{0,0} & \cdots & a_{0,N-1} \\ a_{1,0} & \cdots & a_{1,N-1} \\ \vdots & \ddots & \vdots \\ a_{N-1,0} & \cdots & a_{N-1,N-1} \end{bmatrix}$$

donde cada elemento $a_{i,j}$ representa el coste de energía del sector en las coordenadas (i, j) dentro de la cuadrícula. Cada posición corresponde a una ciudad y su valor indica la energía necesaria para permanecer o desplazarse desde ese punto.

Además, se recibe una lista:

$$A = [a_0, a_1, \ldots, a_{F-1}] \qquad \text{tal que} \qquad \forall\, a_k \in A,\ a_k = (x_k, y_k)$$

donde cada a_k representa la posición inicial del líder k dentro de la matriz M.

2.2. Salida

Si existe al menos una casilla $M[r][c]$ alcanzable por *todos* los F líderes en $\leq T$ movimientos, devuelve la pareja (x', y') y X, tal que $M[x'][y']$ es el sector optimo que minimiza el **costo total** de los F líderes, hasta una de las casillas alcanzables, y donde X es el **costo total** hasta el sector (x', y')

- Si no existe ninguna casilla alcanzable por todos los líderes cumpliendo T, devolver -1

2.3. Cálculo de Coste total

Sea un líder que parte de (x_i, y_i) y llega a una casilla sede (r, c) mediante un camino con L movimientos.

- El coste del viaje de ese líder es la suma de los costes $c_{u,v}$ de todas las casillas que atraviesa, *incluyendo la casilla de origen* (x_i, y_i) y todas las intermedias, pero *excluyendo la casilla destino* (r, c).
- El camino elegido debe cumplir $L \leq T$. Siendo L la cantidad mínima de pasos hasta la casilla (r, c). Si para algún líder no existe un camino con $L \leq T$ hasta (r, c), entonces la casilla (r, c) no es válida.

Así, el coste total de reunir a todos los líderes en la sede (r, c) se define como:

$$X(r, c) = \sum_{i=1}^{F} \text{CosteMínimoExcluyendoDestino}(x_i, y_i, r, c)$$

donde CosteMínimoExcluyendoDestino representa la mínima suma posible de los costes de energía del recorrido desde el líder i hasta (r, c), cumpliendo la restricción de movimientos y *sin contar el coste del sector anfitrión.*

3. Casos de Prueba

3.1. Caso 1: Varias soluciones con el mismo costo

Descripción: Este caso representa una situación sencilla donde existen varias opciones óptimas con el mismo costo total. El propósito es mostrar cómo el algoritmo resuelve empates utilizando el criterio de orden lexicográfico, seleccionando el sector con las coordenadas mayores cuando hay más de una solución igualmente buena.

Entrada:

```
4 2 2
4 1 4 6
1 1 2 5
1 1 1 2
4 6 11 7
1 1 3 1
0 0 0
```

Resultado esperado:

```
The Galactic Summit will be held at sector (3,1) with total energy cost = 2
```

Explicación: Tenemos una cuadrícula de 3×3 con los siguientes costos de energía:

$$\begin{bmatrix} 1 & 2 & 4 \\ 1 & 1 & 2 \\ 1 & 3 & 1 \end{bmatrix}$$

Los dos líderes se encuentran en las posiciones: - Líder 1: $(1,1)$ - Líder 2: $(3,1)$

Cada líder puede moverse hasta $T = 2$ pasos. Con $T = 2$ las posiciones alcanzables son:

- Desde $(1,1)$: $(1,1), (1,2), (1,3), (2,1), (2,2), (3,1)$.
- Desde $(3,1)$: $(3,1), (3,2), (3,3), (2,1), (2,2), (1,1)$.

Ambos líderes pueden reunirse en distintas celdas dentro de su rango de movimiento, y hay más de una celda donde el costo total es mínimo. Específicamente, las celdas $(1,1)$, $(2,1)$, $(2,2)$ y $(3,1)$ generan el mismo costo total mínimo.

Para romper el empate, se selecciona el sector con el **mayor orden lexicográfico**, es decir, aquel cuya fila o columna sea mayor cuando los costos son iguales. Por tanto, entre $(1,1)$, $(2,1)$, $(2,2)$ y $(3,1)$, el algoritmo elige $(3,1)$.

Salida:

$$(3,1), \quad 2$$

Razonamiento del resultado: - Desde $(1,1)$ a $(3,1)$, el líder 1 puede llegar en 2 movimientos (por ejemplo, $(1,1) \to (2,1) \to (3,1)$) con un costo acumulado de $1 + 1 = 2$. Porque, dado que la ultima ciudad es la host, el costo de esa ciudad no se cuenta. - Desde $(3,1)$ a $(3,1)$, el líder 2 gasta 0. - Costo total $= 2 + 0 = 2$.

3.2. Caso 2: Único candidato

Descripción: Caso en el que, debido a la restricción de movimientos T, sólo existe un sector que es alcanzable por *todos* los líderes; por tanto, la solución es única.

Entrada:

```
4 2 1
5 1 5 3
3 4 17 6
10 8 6 7
4 11 9 2
1 1 1 3
0 0 0
```

Resultado esperado:

```
The Galactic Summit will be held at sector (1,2) with total energy cost = 10
```

Explicación: La cuadrícula 3×3 es

$$\begin{bmatrix} 5 & 1 & 5 & 3 \\ 3 & 4 & 17 & 6 \\ 10 & 8 & 6 & 7 \\ 4 & 11 & 9 & 2 \end{bmatrix}$$

Los líderes están en: - Líder 1: $(1, 1)$ - Líder 2: $(1, 3)$

Cada líder puede moverse a lo sumo $T = 1$ pasos. Con $T = 1$ las posiciones alcanzables son:

- Desde $(1, 1)$: $(1, 1), (1, 2), (2, 1)$.
- Desde $(1, 3)$: $(1, 3), (1, 2), (2, 3)$.

La intersección (sectores alcanzables por ambos) es únicamente $(1, 2)$. Por tanto, solo hay un candidato válido y se debe elegir $(1, 2)$.

Cálculo del coste total siguiendo la especificación: - Camino mínimo desde $(1, 1)$ hasta $(1, 2)$ en $\leq T$ movimientos: $(1, 1) \to (1, 2)$. Coste aportado = coste de la casilla de origen $= M[1][1] = 5$. - Camino mínimo desde $(1, 3)$ hasta $(1, 2)$ en $\leq T$ movimientos: $(1, 3) \to (1, 2)$. Coste aportado = coste de la casilla de origen $= (M[1][3]) = 5$.

Costo total $X = 5 + 5 = 10$.

Salida:

$$(1, 2), \quad 10$$

3.3. Caso 3: Sin solución posible

Descripción: Este caso ilustra una situación en la que, debido a la distribución de los líderes y el límite de movimientos T, **no existe ninguna ciudad que sea alcanzable por todos los líderes**. El propósito es mostrar cómo el algoritmo debe identificar correctamente que no hay intersección entre las posiciones accesibles de los líderes y, por tanto, devolver -1.

Entrada:

```
4 2 2
2 4 1 6
5 7 12 3
18 21 9 4
3 4 1 13
1 4 4 1
0 0 0
```

Resultado esperado:

```
Zlatan is disappointed
```

Explicación: La cuadrícula 4×4 es:

$$\begin{bmatrix} 2 & 4 & 1 & 6 \\ 5 & 7 & 12 & 3 \\ 18 & 21 & 9 & 4 \\ 3 & 4 & 1 & 13 \end{bmatrix}$$

Los líderes se encuentran en: - Líder 1: $(1,4)$ - Líder 2: $(4,1)$

Cada líder puede moverse a lo sumo $T = 2$ pasos. Con ese límite, las posiciones alcanzables son:

- Desde $(1,4)$: $(1,4)$, $(1,3)$, $(2,4)$, $(1,2)$, $(2,3)$, $(3,4)$. - Desde $(4,1)$: $(4,1)$, $(3,1)$, $(4,2)$, $(2,1)$, $(3,2)$, $(4,3)$.

Al comparar ambos conjuntos, se observa que **no hay ninguna intersección**. Es decir, no existe ninguna casilla del imperio a la que ambos líderes puedan llegar respetando el número máximo de movimientos $T = 2$.

Por tanto, el problema no tiene solución válida para este escenario.

Salida:

$$-1$$

4. Explicación general de la estrategia de solución

4.1. Primera estrategia de solución

La idea inicial consiste en diseñar un algoritmo (aún en proceso de definición) que reciba como entrada la matriz completa de costos de energía, la posición inicial de todos los líderes y el número máximo de pasos T que cada uno puede realizar.

El objetivo del algoritmo será determinar todas las coordenadas (r, c) que puedan ser alcanzadas por **todos** los líderes utilizando como máximo T movimientos.

Una vez identificadas las posiciones candidatas, se ejecutará un algoritmo de búsqueda de caminos mínimos (*Dijkstra*) sobre el grafo implícito definido por la matriz. Este proceso se aplicará desde cada líder hacia todas las posiciones candidatas, con el fin de obtener el **camino de costo mínimo** cumpliendo la restricción de $L \leq T$.

Para calcular el **costo total**, se considera que un líder parte desde (x_i, y_i) y llega a una posible sede (r, c) mediante un camino de longitud L. El costo individual del viaje de dicho líder será la suma de los costos de las casillas recorridas, incluyendo la casilla de origen e intermedias, pero **excluyendo la casilla destino**.

Formalmente, el costo total de reunir a todos los líderes en una sede (r, c) se define como:

$$X(r,c) = \sum_{i=1}^{F} \text{CosteMínimoExcluyendoDestino}(x_i, y_i, r, c)$$

donde CosteMínimoExcluyendoDestino representa la mínima suma posible de energía del recorrido desde el líder i hasta (r, c), cumpliendo la restricción de movimientos $L \leq T$.

Si para algún líder no existe un camino válido hasta (r, c), esa casilla se descarta como posible sede. Finalmente, se evalúan todas las posiciones candidatas válidas y se selecciona aquella con el **menor costo total** $X(r, c)$.

Aspectos a perfeccionar

- Determinar el método más eficiente para generar las posiciones candidatas alcanzables por todos los líderes sin necesidad de exploración redundante.
- Analizar cómo limitar la búsqueda a T pasos sin afectar la exactitud de los caminos mínimos.

4.2. Estrategia final de solución

La solución definitiva retoma la idea inicial de usar caminos mínimos, pero la concreta en un esquema sistemático que combina tres componentes principales:

1. Modelar la cuadrícula como un grafo implícito con pesos en los vértices.
2. Ejecutar, para cada líder, un algoritmo de Dijkstra acotado a un máximo de T movimientos.
3. Agregar de forma global la información de costos y alcanzabilidad, para elegir finalmente la sede óptima con un criterio de desempate bien definido.

Modelado del problema como grafo

Cada sector del imperio es una celda (i, j) de la matriz de costos de energía. Estas celdas se interpretan como vértices de un grafo no dirigido $G = (V, E)$, donde:

- Cada vértice representa un sector (i, j) de la cuadrícula.
- Hay una arista entre dos vértices si los sectores correspondientes son adyacentes en una de las cuatro direcciones cardinales (norte, sur, este, oeste).

A diferencia de modelos donde el peso se asigna a las aristas, aquí el costo de un camino se define como la suma de los costos de las celdas visitadas, incluyendo la celda de origen y todas las intermedias, pero excluyendo la celda sede. En términos prácticos, esto se maneja manteniendo, para cada líder, una matriz de *costo mínimo acumulado* hasta cada celda, y descontando el costo de la celda sede únicamente al momento de sumar el costo global de una posible reunión.

Cálculo de rutas mínimas para cada líder

Para cada líder se calcula, de manera independiente, el costo mínimo de energía para llegar a cada celda de la cuadrícula, sujeto a la restricción de no usar más de T movimientos. Para ello se utiliza una variante del algoritmo de Dijkstra sobre el grafo implícito descrito anteriormente.

La idea es la siguiente:

1. Para un líder situado en (x_i, y_i), se inicializa una matriz de costos mínimos con valor infinito en todas las celdas, salvo en la posición inicial, donde el costo se fija igual al costo de energía del sector de partida (pues el líder "paga" su ciudad de origen).

2. Se utiliza una estructura de datos tipo **cola de prioridad** que almacena estados de la forma:

$$(\text{número de pasos}, \text{costo acumulado}, \text{fila}, \text{columna}),$$

ordenados primero por el número de pasos y, en caso de empate, por el costo acumulado. Esto asegura que:

   - Nunca se continúa expandiendo un estado que ya ha superado el límite de pasos T.
   - Entre estados con la misma cantidad de pasos, se propaga primero aquel que lleva menor costo acumulado.

3. Mientras la cola de prioridad no esté vacía, se extrae el estado con mejor prioridad. Si ese estado no mejora el mejor costo conocido para su celda, se descarta. En caso contrario:

   - Si el número de pasos usados es menor que T, se generan los vecinos alcanzables (arriba, abajo, izquierda, derecha), siempre que estén dentro de la cuadrícula.
   - Para cada vecino válido se calcula el nuevo costo como

   $$\text{costo nuevo} = \text{costo actual} + \text{costo del vecino},$$

   y el número de pasos aumenta en una unidad. Si este nuevo costo mejora el mejor costo conocido para esa celda vecina, se actualiza la matriz de costos y se inserta el nuevo estado en la cola de prioridad.

Al finalizar este proceso para un líder concreto, se dispone de una matriz que indica, para cada sector alcanzable en $\leq T$ movimientos, cuál es el costo mínimo de energía que ese líder tendría que invertir para llegar allí.

Agregación global de costos y verificación de alcanzabilidad

Una vez se han calculado las matrices de costos para todos los líderes, se combinan sus resultados mediante una estructura de agregación global (conceptualmente, un *mapa* o diccionario) cuya clave es la coordenada (r, c) de un sector y cuyo valor almacena dos datos:

- Un **marcador de alcanzabilidad**, que acumula la "contribución" de cada líder que logra llegar a ese sector.
- Un **costo total acumulado**, que suma los costos mínimos individuales de todos los líderes que pueden alcanzar esa coordenada, siempre excluyendo el costo del propio sector sede.

El procedimiento para combinar la información de cada líder es:

- Se recorre la matriz de costos mínimos de ese líder. Para cada celda (r, c) cuya entrada no sea infinita, se sabe que el líder puede llegar a ese sector dentro del límite de movimientos.

- Para esa celda se actualiza el marcador de alcanzabilidad sumando un identificador asociado a ese líder (por ejemplo, un número único del 1 al F).
- Además, se suma al costo total acumulado el costo mínimo desde la posición del líder hasta (r, c), pero descontando el costo del propio sector sede, dado que ese costo lo asume Zlatan.

Si se asume que los líderes tienen identificadores $1, 2, \ldots, F$, entonces la suma esperada de identificadores, en caso de que *todos* los líderes alcancen una celda, es:

$$\text{SumaEsperada} = \sum_{i=1}^{F} i = 1 + 2 + \cdots + F = \frac{F(F+1)}{2}.$$

Una coordenada (r, c) se considera candidata válida a sede si, y sólo si, el marcador de alcanzabilidad en esa celda coincide con SumaEsperada. En otras palabras, sólo se tienen en cuenta como posibles sedes aquellas ciudades a las que todos los líderes pueden llegar respetando el límite de T movimientos.

Selección de la sede óptima y criterio de desempate

La sede final del "Galactic Summit" se elige recorriendo todas las celdas candidatas válidas (es decir, aquellas cuyo marcador de alcanzabilidad indica que son accesibles por todos los líderes):

1. Entre todas las celdas candidatas (r, c), se selecciona inicialmente la que tenga el **menor costo total acumulado** de energía.
2. Si existe más de una celda con el mismo costo total mínimo, se aplica un criterio de desempate basado en el orden lexicográfico de las coordenadas:
   - Se elige la celda con la **mayor fila** r.
   - Si hay empate en la fila, se elige la que tenga la **mayor columna** c.

   De este modo, cuando existen varias sedes igualmente buenas en términos de costo, se escoge la coordenada "más grande" en el sentido del orden lexicográfico.

Si al final del proceso no existe ninguna celda que haya sido marcada como alcanzable por todos los líderes, se concluye que no hay una ciudad en la que todos puedan reunirse cumpliendo el límite de movimientos T, y el problema no tiene solución: en ese caso, el resultado corresponde al mensaje de que Zlatan queda decepcionado.

5. Desarrollo paso a paso de un caso de prueba representativo

Para ilustrar el comportamiento de la estrategia final en un escenario más cercano a los tamaños del problema real, se presenta un caso de prueba con:

- Una cuadrícula de tamaño $N = 5$.
- $F = 3$ líderes.

- Un máximo de $T = 4$ movimientos por líder.
- Múltiples ciudades candidatas con el mismo costo total mínimo, lo que obliga a aplicar el criterio de desempate lexicográfico.

5.1. Sample input y expected output (formato UVA)

En el formato del problema, esta instancia se puede escribir como:

```
5 3 4
4 8 1 4 9
7 9 4 7 6
1 2 2 2 5
4 3 3 4 9
7 4 1 4 8
4 3 1 5 2 1
0 0 0
```

donde:

- La primera línea indica $N = 5$, $F = 3$, $T = 4$.
- Las siguientes 5 líneas son la matriz de costos de energía.
- La línea `4 3 1 5 2 1` contiene las posiciones iniciales de los tres líderes:

$$(4, 3), \ (1, 5), \ (2, 1).$$

- La línea `0 0 0` indica el fin de la entrada.

La salida esperada para este caso es:

```
The Galactic Summit will be held at sector (3,3) with total energy cost = 31
```

5.2. Descripción de la instancia

La matriz de costos de energía del imperio es:

$$M = \begin{bmatrix} 4 & 8 & 1 & 4 & 9 \\ 7 & 9 & 4 & 7 & 6 \\ 1 & 2 & 2 & 2 & 5 \\ 4 & 3 & 3 & 4 & 9 \\ 7 & 4 & 1 & 4 & 8 \end{bmatrix}$$

Las posiciones iniciales de los líderes (en coordenadas $(\text{fila}, \text{columna})$, indexadas desde 1) son:

- Líder 1: $(4, 3)$.
- Líder 2: $(1, 5)$.
- Líder 3: $(2, 1)$.

Cada líder puede realizar como máximo $T = 4$ movimientos en las cuatro direcciones cardinales. Buscamos un sector (r, c) al que los tres líderes puedan llegar respetando ese límite de movimientos, minimizando el costo total de energía, excluyendo siempre el costo del sector sede.

5.3. Paso 1: cálculo de rutas mínimas para cada líder

Aplicando el algoritmo de Dijkstra acotado a $T = 4$ movimientos desde la posición inicial de cada líder, se obtiene para cada uno una matriz de **costo mínimo acumulado** hasta cada celda (incluyendo el costo de la celda de origen y de todas las intermedias). Las celdas inalcanzables dentro de los T pasos se marcan como ∞.

Líder 1 en $(4, 3)$

La matriz de costos mínimos es:

$$C^{(1)} = \begin{bmatrix} \infty & 18 & 10 & 14 & \infty \\ 15 & 16 & 9 & 14 & 18 \\ 8 & 7 & 5 & 7 & 12 \\ 10 & 6 & 3 & 7 & 16 \\ 15 & 8 & 4 & 8 & 16 \end{bmatrix}$$

Por ejemplo, el valor $C^{(1)}_{3,3} = 5$ indica que el costo mínimo para que el Líder 1 llegue al sector $(3, 3)$ es 5 unidades de energía (sumando origen e intermedias).

Líder 2 en $(1, 5)$

La matriz de costos mínimos es:

$$C^{(2)} = \begin{bmatrix} 26 & 22 & 14 & 13 & 9 \\ \infty & 27 & 18 & 20 & 15 \\ \infty & \infty & 20 & 22 & 20 \\ \infty & \infty & \infty & 26 & 29 \\ \infty & \infty & \infty & \infty & 37 \end{bmatrix}$$

Por ejemplo, $C^{(2)}_{2,3} = 18$ indica que el costo mínimo para que el Líder 2 llegue a $(2, 3)$ es 18 unidades de energía.

Líder 3 en $(2, 1)$

La matriz de costos mínimos es:

$$C^{(3)} = \begin{bmatrix} 11 & 19 & 20 & 24 & \infty \\ 7 & 16 & 16 & 27 & 33 \\ 8 & 10 & 12 & 14 & \infty \\ 12 & 13 & 15 & \infty & \infty \\ 19 & 17 & \infty & \infty & \infty \end{bmatrix}$$

Por ejemplo, $C^{(3)}_{3,3} = 12$ indica que el costo mínimo para que el Líder 3 llegue a $(3, 3)$ es 12 unidades de energía.

5.4. Paso 2: sectores alcanzables por todos los líderes

Un sector (r, c) es candidato a sede solamente si es alcanzable por *todos* los líderes dentro de los T movimientos; es decir, si $C^{(k)}_{r,c}$ es finito para cada líder k.

A partir de las tres matrices de costos mínimos, los sectores que cumplen esta condición son:

$$\{(1,2),(1,3),(1,4),(2,2),(2,3),(2,4),(2,5),(3,3),(3,4)\}.$$

Todos estos sectores son accesibles en ≤ 4 movimientos por los tres líderes y, por tanto, son las únicas sedes posibles.

5.5. Paso 3: cálculo del costo total para cada candidata

Recordemos que el costo global de reunir a todos los líderes en un sector (r,c) se define como:

$$X(r,c) = \sum_{k=1}^{F} \left(C_{r,c}^{k} - M_{r,c} \right),$$

ya que cada $C_{r,c}^{(k)}$ incluye el costo del sector sede $M_{r,c}$, y este debe excluirse de la suma total (lo asume Zlatan, no los líderes).

Aplicando esta fórmula a cada sector candidato se obtiene:

Sector (r,c)	$X(r,c)$
$(1,2)$	35
$(1,3)$	41
$(1,4)$	39
$(2,2)$	32
$(2,3)$	31
$(2,4)$	40
$(2,5)$	48
$(3,3)$	31
$(3,4)$	37

Se observa que el **costo total mínimo** es $X = 31$, alcanzado en dos sectores distintos:

$$(2,3) \quad \text{y} \quad (3,3).$$

Para ilustrar el cálculo en detalle, consideremos estos dos sectores:

Sector $(2,3)$. Aquí el costo del sector sede es $M_{2,3} = 4$. Las contribuciones de cada líder son:

Líder	$C_{2,3}^{k}$	$C_{2,3}^{k} - M_{2,3}$
1	9	$9-4=5$
2	18	$18-4=14$
3	16	$16-4=12$

Por tanto,

$$X(2,3) = 5 + 14 + 12 = 31.$$

Sector $(3,3)$. Aquí el costo del sector sede es $M_{3,3} = 2$. Las contribuciones son:

Líder	$C_{3,3}$	$C_{3,3} - M_{3,3}$
1	5	$5-2=3$
2	20	$20-2=18$
3	12	$12-2=10$

Luego,

$$X(3,3) = 3 + 18 + 10 = 31.$$

Ambos sectores son igualmente buenos en términos de costo total.

5.6. Paso 4: aplicación del criterio de desempate

Dado que existe un empate entre $(2,3)$ y $(3,3)$ con el mismo costo total mínimo $X = 31$, se aplica el criterio de desempate definido en la estrategia:

1. Se elige el sector con la **mayor fila** r.
2. Si las filas coinciden, se elige el de la **mayor columna** c.

Entre las dos opciones:

$$(2,3) \quad \text{y} \quad (3,3),$$

la que tiene mayor fila es $(3,3)$. Por tanto, el sector seleccionado como sede del *Zlatan Galactic Summit* es:

$$(r,c) = (3,3),$$

con costo total de energía

$$X(3,3) = 31.$$

5.7. Conclusión del caso representativo

Este caso de prueba muestra cómo la estrategia final:

- Calcula rutas de costo mínimo para cada líder respetando el límite máximo de movimientos T.
- Determina los sectores alcanzables por todos los líderes.
- Evalúa el costo total de reunir a todos en cada sector candidato, excluyendo el costo del anfitrión.
- Resuelve empates mediante un criterio lexicográfico sobre las coordenadas, eligiendo finalmente el sector $(3,3)$ como sede óptima.

Aunque el ejemplo utiliza una cuadrícula 5×5, el mismo procedimiento se aplica de manera análoga para instancias de mayor tamaño, manteniendo la misma lógica de cálculo y selección.

6. Cambios entre la primera versión y la entrega final

En esta sección se describen los principales cambios realizados entre la primera versión de la solución (entrega 1) y la versión final presentada en este informe. Los ajustes se dieron tanto a nivel conceptual como algorítmico y de validación mediante casos de prueba.

6.1. Clarificación del modelo de costos

En la primera versión la definición del costo del recorrido no estaba completamente formalizada en términos del modelo de grafo. En particular:

- No se explicitaba con suficiente precisión que el costo total de un camino se obtiene sumando los costos de las celdas de origen e intermedias.
- Tampoco se detallaba de forma rigurosa la exclusión del costo de la celda sede en el cálculo global.

En la versión final:

- Se formalizó el modelo como un grafo con pesos en los vértices.
- Se dejó claramente establecido que el costo del sector anfitrión se resta al final para cada líder, ya que lo asume Zlatan y no debe formar parte de la suma que se minimiza.

Esto permitió alinear por completo la implementación con la especificación matemática del problema.

6.2. De un conjunto de candidatos explícito a una agregación global

En la estrategia inicial se planteaba, a nivel conceptual, la idea de:

1. Determinar primero las posiciones candidatas alcanzables por todos los líderes.
2. Ejecutar luego algoritmos de caminos mínimos desde cada líder hacia dichas posiciones candidatas.

Este enfoque dejaba abierta la cuestión de cómo construir el conjunto de candidatos de forma eficiente, sin exploraciones redundantes.

En la versión final, en lugar de mantener un conjunto explícito de candidatos desde el inicio, se adoptó el siguiente esquema:

- Para cada líder se ejecuta *una sola vez* el algoritmo de Dijkstra acotado a T movimientos sobre toda la cuadrícula.
- Los resultados de cada líder se combinan en una estructura global (un mapa o diccionario) que, para cada celda, acumula:
  - Un marcador de alcanzabilidad (que indica qué líderes llegaron a esa celda).
  - El costo total acumulado de traer a todos los líderes que lograron llegar a ella.
- Las celdas candidatas válidas surgen automáticamente como aquellas cuyo marcador de alcanzabilidad indica que *todos* los líderes las alcanzan.

Este cambio simplifica el diseño y evita la necesidad de construir y mantener explícitamente un conjunto de posiciones candidatas desde el principio.

6.3. Control explícito del límite de movimientos T

En la primera entrega la restricción de movimientos T se mencionaba de forma conceptual, pero no se detallaba con claridad cómo incorporarla dentro del algoritmo de caminos mínimos.

En la versión final:

- Se introdujo el número de pasos como parte del estado manejado por el algoritmo de Dijkstra.
- Se estableció que los estados que superan T movimientos no se expanden, garantizando que todas las distancias calculadas corresponden a caminos de longitud $\leq T$.
- Se describió explícitamente el orden de prioridad en la exploración (primero por número de pasos y luego por costo), asegurando la corrección del algoritmo bajo esta restricción.

De esta forma, la limitación de movimientos deja de ser una condición abstracta y pasa a estar integrada de forma directa en la lógica del algoritmo.

6.4. Definición precisa del criterio de desempate

En la primera versión el tratamiento de empates entre diferentes posibles sedes con el mismo costo total no estaba completamente desarrollado.

En la solución final:

- Se adoptó un criterio de desempate claro y consistente: entre todas las celdas con el mismo costo mínimo, se elige aquella con mayor fila, y en caso de empate en la fila, la de mayor columna.
- Se documentó este criterio tanto en la especificación informal como en el desarrollo del caso de prueba representativo, mostrando ejemplos concretos donde varias sedes comparten el mismo costo global.

Esto asegura que, ante cualquier empate, la respuesta del algoritmo es determinista y está justificada en el informe.

6.5. Mejoras en los casos de prueba y en la documentación

Finalmente, entre la entrega inicial y la versión final se reforzó de forma importante la parte de validación y documentación:

- Se mantuvieron y detallaron los casos sencillos de la entrega 1 (varias soluciones óptimas, único candidato y caso sin solución), explicitando las trayectorias y los costos.
- Se añadió un caso de prueba más amplio y representativo con una matriz de tamaño 5×5 y tres líderes, donde se muestran:
  - Las matrices de costos mínimos para cada líder.
  - La intersección de sectores alcanzables.

- El cálculo detallado del costo total y la resolución de empates.

- Se reescribió la explicación de la estrategia en dos niveles: una *primera estrategia de solución* (correspondiente a la entrega inicial) y una *estrategia final de solución*, que describe con precisión el enfoque definitivo implementado.

Estos cambios no modifican la esencia del problema, pero sí mejoran la claridad, la trazabilidad entre especificación y código, y la evidencia de corrección de la solución propuesta.

7. Análisis de complejidad de la solución

En esta sección se analiza la complejidad temporal y espacial del algoritmo propuesto, en función de los parámetros del problema:

- N: tamaño de la cuadrícula ($N \times N$ sectores).
- F: número de líderes.
- T: número máximo de movimientos permitidos por líder.

Para simplificar, denotaremos por $V = N^2$ el número de ciudades (vértices del grafo) y por E el número de conexiones entre ellas. En una cuadrícula con movimientos a las cuatro direcciones cardinales se tiene $E = \Theta(N^2)$, ya que cada ciudad tiene a lo sumo 4 vecinos.

7.1. Complejidad temporal

La solución se puede descomponer en tres fases principales por instancia de entrada:

1. Lectura de la matriz de costos y de las posiciones de los líderes.
2. Cálculo de rutas mínimas para cada líder (ejecuciones de Dijkstra acotadas por T).
3. Agregación global de resultados y selección de la sede óptima.

1. Lectura de datos

La lectura de la matriz de costos de tamaño $N \times N$ tiene coste $\mathcal{O}(N^2)$. La lectura de las F posiciones iniciales de los líderes tiene coste $\mathcal{O}(F)$.

En conjunto, esta fase es de orden:

$$\mathcal{O}(N^2 + F).$$

2. Algoritmo de Dijkstra para cada líder

Para cada líder se ejecuta una variante del algoritmo de Dijkstra sobre el grafo implícito de la cuadrícula, con las siguientes características:

- El grafo tiene $V = N^2$ vértices y $E = \Theta(N^2)$ aristas (movimientos arriba, abajo, izquierda, derecha).
- Se utiliza una cola de prioridad para seleccionar el siguiente estado a expandir.
- El estado incluye tanto el costo acumulado como el número de pasos usados, y sólo se expanden estados cuyo número de pasos es estrictamente menor que T.

En el peor caso, si T es suficientemente grande, el recorrido puede alcanzar todas las celdas de la cuadrícula. Cada ciudad puede ser insertada en la cola de prioridad un número acotado de veces, ya que sólo se actualiza cuando se encuentra un costo estrictamente menor que el actual. Por tanto, el número total de operaciones sobre la cola de prioridad queda acotado por un múltiplo de V y E.

La complejidad clásica de Dijkstra usando cola de prioridad binaria sobre un grafo con V vértices y E aristas es:

$$\mathcal{O}\big((V + E)\log V\big).$$

En nuestro caso, como $V = \Theta(N^2)$ y $E = \Theta(N^2)$, se obtiene:

$$\mathcal{O}\big((N^2 + N^2)\log(N^2)\big) = \mathcal{O}\big(N^2 \log N\big)$$

para una ejecución de Dijkstra asociada a un único líder.

Este procedimiento se repite para los F líderes, de modo que el coste total de esta fase es:

$$\mathcal{O}\big(F \cdot N^2 \log N\big).$$

El límite de movimientos T actúa como una poda: si T es pequeño, el número de celdas realmente exploradas se reduce y, por tanto, el coste efectivo puede ser menor. Sin embargo, en el análisis asintótico de peor caso se asume que T es lo suficientemente grande como para permitir alcanzar todas las celdas relevantes, por lo que la cota se mantiene en $\mathcal{O}(FN^2 \log N)$.

3. Agregación de resultados y elección de la sede

Tras ejecutar Dijkstra para cada líder, se realiza:

- Un recorrido completo de la cuadrícula por líder para actualizar la estructura global de agregación (mapa o diccionario que asocia a cada sector su marcador de alcanzabilidad y su costo total acumulado).
- Una pasada final sobre esa estructura para seleccionar la mejor sede según el costo total y el criterio de desempate lexicográfico.

Recorridos por líder. Para cada líder, se recorren las N^2 celdas de la matriz de costos mínimos. En cada celda se realiza una actualización sobre la estructura global, cuya operación básica tiene coste $\mathcal{O}(\log K)$, donde K es el número de entradas almacenadas (a lo sumo $V = N^2$).

Por tanto, el costo de esta etapa para un líder es:

$$\mathcal{O}\big(N^2 \log N^2\big) = \mathcal{O}\big(N^2 \log N\big),$$

y para los F líderes:

$$\mathcal{O}\big(F \cdot N^2 \log N\big).$$

Selección final de la sede. La estructura global puede llegar a contener, en el peor caso, información para todas las celdas de la cuadrícula, es decir, $\mathcal{O}(N^2)$ entradas. La selección de la sede óptima se hace recorriendo linealmente dicha estructura y aplicando comparaciones constantes por entrada (costo total y criterio de desempate sobre las coordenadas).

El coste de esta fase es, por tanto:

$$\mathcal{O}(N^2).$$

Complejidad temporal total

Sumando las contribuciones dominantes de cada fase, se obtiene que la complejidad temporal de la solución para una instancia es:

$$\mathcal{O}(N^2 + F + FN^2 \log N + FN^2 \log N + N^2) = \mathcal{O}\big(FN^2 \log N\big).$$

Por tanto, el tiempo de ejecución crece de manera casi lineal con el número de líderes F y cuadrática (con un factor logarítmico) con el tamaño N de la cuadrícula.

7.2. Complejidad espacial

En cuanto al uso de memoria, los componentes principales son:

- La matriz de costos del imperio, de tamaño $N \times N$: $\mathcal{O}(N^2)$.
- La matriz de costos mínimos utilizada por Dijkstra para cada líder: $\mathcal{O}(N^2)$.
- La estructura global de agregación (mapa o diccionario) que almacena, para cada sector, el marcador de alcanzabilidad y el costo total acumulado: en el peor caso, puede contener una entrada por cada sector, es decir, $\mathcal{O}(N^2)$.
- La estructura auxiliar para las posiciones de los líderes, de tamaño $\mathcal{O}(F)$.

La memoria consumida por la cola de prioridad de Dijkstra y por otras estructuras auxiliares es proporcional al número de celdas visitadas, y está acotada por $\mathcal{O}(N^2)$.

En consecuencia, la complejidad espacial total de la solución es:

$$\mathcal{O}(N^2 + F),$$

dominada por los términos cuadráticos asociados a las matrices y a la estructura global de resultados. En la práctica, el término $\mathcal{O}(N^2)$ es el relevante, puesto que el número de líderes F suele ser considerablemente menor que el número total de sectores del imperio.