

Articulo "Codigo Python"

Licenciatura en Ciencias de la Computacion
Pontificia Univesidad Catolica de Chile

Integrantes: Constanza Ocaranza y Camila Riquelme

1 RESOLUCCION DEL PROBLEMA DE LA MATRIZ 5X5

DESCRIPCION : Se nos presenta una matriz de 5x5 en la que 24 de sus elementos son ceros y uno de ellos es un 1. El objetivo es trasladar ese único número 1 desde su posición original hasta la posición central de la matriz, que corresponde a la intersección de la tercera fila y la tercera columna.

1.1 De donde salio

este codigo fue echo por mi querida socia camilita

1.2 EL RETO

consiste en determinar la cantidad mínima de movimientos necesarios para trasladar ese número 1 al centro de la matriz, utilizando las siguientes transformaciones permitidas:

1. -Intercambiar dos filas vecinas: Es decir,intercambiar una fila con la siguiente, por ejemplo, la fila 2 con la fila 3.
2. -Intercambiar dos columnas vecinas: Similar a las filas, se pueden intercambiar dos columnas adyacentes, como la columna 3 con la columna 4.

El problema se resuelve con el mínimo número de movimientos posibles, y se nos pide que calculemos este número.

FORMATO DE ENTRADA Y SALIDA

1.3 ENTRADA

La entrada consiste en cinco líneas, cada una de ellas con cinco números enteros. Los números representan los elementos de la matriz de 5x5. En esta matriz, habrá 24 ceros y exactamente un número 1. La posición de este número 1 es aleatoria dentro de la matriz.

1.4 SALIDA

La salida será un único número entero, el cual representará la cantidad mínima de movimientos necesarios para mover el número 1 a la posición central de la matriz (en la intersección de la fila 3 y la columna 3).

2 ANALISIS DEL PROBLEMA

El problema se basa en **distancias**: la cantidad mínima de movimientos necesarios es directamente proporcional a la distancia entre la posición inicial del número 1 y la posición central.

Para cada fila y columna, el número de movimientos necesarios para colocar el 1 en el centro es simplemente la **diferencia absoluta** entre las posiciones actuales y las posiciones deseadas.

2.1 POSICION DEL PROBLEMA

La posición central de la matriz es la intersección de la **tercera fila** y la tercera columna, que tiene la coordenada (3, 3) en un índice basado en 1.

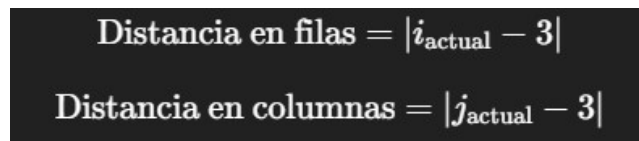

$$\text{Distancia en filas} = |i_{\text{actual}} - 3|$$
$$\text{Distancia en columnas} = |j_{\text{actual}} - 3|$$

Figure 1: Calculo de movimiento

2.2 POSICION INICIAL

La posición inicial del número 1 se debe encontrar buscando su ubicación dentro de la matriz. Esta búsqueda puede hacerse recorriendo la matriz fila por fila y columna por columna hasta encontrar el número 1.

2.3 CALCULO DE MOVIMIENTO

Para calcular el número de movimientos necesarios para mover el 1 a la posición central, basta con obtener las distancias entre las posiciones actuales del número 1 y la posición (3, 3). Estas distancias se calculan como la diferencia absoluta en las filas y las columnas: (El número total de movimientos es la suma de estas distancias.)

2.4 ALGORITMO PROPUESTO

Para resolver el problema, el algoritmo sigue los siguientes pasos:

- Leer la matriz: Se leen las 5 líneas de entrada y se almacenan en una estructura adecuada (como una lista bidimensional).
- Encontrar la posición del 1: Recorremos la matriz hasta encontrar las coordenadas (i,j) del número 1.
- Calcular la distancia: Se calcula la diferencia absoluta entre las coordenadas del 1 y la posición central (3, 3).
- Imprimir el resultado: La cantidad de movimientos es la suma de las distancias en filas y columnas.

3 CONCLUSION

El problema de mover el número 1 al centro de una matriz 5x5 es sencillo y se resuelve de manera eficiente utilizando el cálculo de distancias absolutas entre las coordenadas actuales del 1 y la posición central de la matriz. Este tipo de problemas refuerza la comprensión de las distancias Manhattan en espacios discretos y el uso de transformaciones simples (intercambio de filas y columnas).

Este enfoque tiene una complejidad constante (1) $O(1)$ porque solo estamos trabajando con una matriz de tamaño fijo 5x5, lo que hace que el algoritmo sea muy eficiente en términos de tiempo y espacio.

4 link

<https://codeforces.com/problemset/problem/263/A>

```
Code Blame 33 lines (20 loc)...
```

```
1  import sys
2  matriz = []
3  aux = 0
4  index = 0
5
6  datos = list(map(int, sys.stdin.read().split()))
7
8  for i in range(5):
9      fila = []
10     for j in range(5):
11         fila.append(datos[index])
12         index += 1
13     matriz.append(fila)
14
15 for i in range(len(matriz)):
16     print(matriz[i])
17
18 #ok ahora a encontrar la posición del 1
19
20 for i in range(5):
21     for j in range(5):
22         if(matriz[i][j] == 1):
23             aux = aux + abs(i-2)
24             aux = aux + abs(j-2)
25
26 print(aux)
27
```

Figure 2: Código Python