

Sumas de prefijos

Miguel Ortiz

Programación Competitiva para ICPC

En arreglos 1D

Tenemos un arreglo a de N elementos

Queremos obtener el valor de la suma:

$$a[l] + a[l + 1] + \dots + a[r]$$

para Q diferentes pares (l, r) que cumplan $1 \leq l \leq r \leq N$.

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3

En arreglos 1D

Tenemos un arreglo a de N elementos

Queremos obtener el valor de la suma:

$$a[l] + a[l + 1] + \dots + a[r]$$

para Q diferentes pares (l, r) que cumplan $1 \leq l \leq r \leq N$.

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3

- Iterar por todos los valores en el rango $[l, r]$ uno por uno y acumular $a[i]$ es $O(N)$

En arreglos 1D

Tenemos un arreglo a de N elementos

Queremos obtener el valor de la suma:

$$a[l] + a[l + 1] + \dots + a[r]$$

para Q diferentes pares (l, r) que cumplan $1 \leq l \leq r \leq N$.

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3

- Iterar por todos los valores en el rango $[l, r]$ uno por uno y acumular $a[i]$ es $O(N)$, por las Q consultas, la complejidad final es $O(NQ)$
- Si $N, Q \leq 10^5$, la solución será muy lenta

En arreglos 1D

- Podemos calcular la suma de cada prefijo del arreglo para responder las consultas
- Guardamos los valores en un arreglo *pre* de tamaño N

En arreglos 1D

- Podemos calcular la suma de cada prefijo del arreglo para responder las consultas
- Guardamos los valores en un arreglo *pre* de tamaño N
- $pre[k] = \sum_{i=1}^k a[i]$

En arreglos 1D

- Podemos calcular la suma de cada prefijo del arreglo para responder las consultas
- Guardamos los valores en un arreglo *pre* de tamaño N
- $pre[k] = \sum_{i=1}^k a[i]$
- Calcular el valor de cada prefijo con un for es $O(N)$, $O(N^2)$ para todos los prefijos

En arreglos 1D

- Podemos calcular la suma de cada prefijo del arreglo para responder las consultas
- Guardamos los valores en un arreglo *pre* de tamaño N
- $pre[k] = \sum_{i=1}^k a[i]$
- Calcular el valor de cada prefijo con un for es $O(N)$, $O(N^2)$ para todos los prefijos
- Se puede calcular todo el arreglo *pre* en $O(N)$ solo aumentando el ultimo valor y re-utilizando la suma del anterior prefijo

En arreglos 1D

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13		

- Ya calculamos las sumas de todos los prefijos hasta $pre[4]$
- $\sum_{i=1}^5 a[i] = (\sum_{i=1}^4 a[i]) + a[5] = pre[4] + a[5]$

En arreglos 1D

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13		

- Ya calculamos las sumas de todos los prefijos hasta $pre[4]$
- $\sum_{i=1}^5 a[i] = (\sum_{i=1}^4 a[i]) + a[5] = pre[4] + a[5]$
- Para calcular todo el arreglo pre en $O(N)$:
 - $pre[1] = a[1]$
 - $pre[i] = pre[i - 1] + a[i]$ para $2 \leq i \leq N$

En arreglos 1D

- Vemos como calcular las consultas utilizando prefijos
- Queremos la suma de los elementos en el rango $[3, 5]$

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13	18	21

En arreglos 1D

- Vemos como calcular las consultas utilizando prefijos
- Queremos la suma de los elementos en el rango [3, 5]

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13	18	21
+						

En arreglos 1D

- Vemos como calcular las consultas utilizando prefijos
- Queremos la suma de los elementos en el rango $[3, 5]$

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13	18	21
+						
-						

En arreglos 1D

- Vemos como calcular las consultas utilizando prefijos
- Queremos la suma de los elementos en el rango $[3, 5]$

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13	18	21
+						
-						
=						

En arreglos 1D

- Vemos como calcular las consultas utilizando prefijos
- Queremos la suma de los elementos en el rango $[3, 5]$

i	1	2	3	4	5	6
a[i]	1	6	4	2	5	3
pre[i]	1	7	11	13	18	21
+						
-						
=						

- $\sum_{i=3}^5 a[i] = pre[5] - pre[2]$

En arreglos 1D

- Para cada consulta (l, r) , obtenemos la respuesta con:

$$pre[r] - pre[l - 1]$$

- El tiempo para responder las Q consultas en un arreglo de tamaño N es $O(N + Q)$

En arreglos 1D

```
int n;  
int a[n];  
int pre[n];  
  
for(int i = 0; i < n; i++) {  
    pre[i] = a[i];  
    if (i > 0) pre[i] += pre[i-1];  
}
```

En matrices 2D

- Podemos aplicar la misma técnica para obtener el valor de una submatriz

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener una matriz $pre[x][y]$ que guarde el valor de $\sum_{i=1}^x \sum_{j=1}^y mat[i][j]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener una matriz $pre[x][y]$ que guarde el valor de $\sum_{i=1}^x \sum_{j=1}^y mat[i][j]$
- Por ejemplo, $pre[2][3] = 11$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener una matriz $pre[x][y]$ que guarde el valor de $\sum_{i=1}^x \sum_{j=1}^y mat[i][j]$
- Por ejemplo, $pre[2][3] = 11$
- También, $pre[4][2] = 13$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener una matriz $pre[x][y]$ que guarde el valor de $\sum_{i=1}^x \sum_{j=1}^y mat[i][j]$
- Por ejemplo, $pre[2][3] = 11$
- También, $pre[4][2] = 13$
- Veremos como calcular la matriz pre en $O(NM)$, siendo N las filas de la matriz original y M las columnas

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i - 1][j] + pre[i][j - 1] - pre[i - 1][j - 1] + mat[i][j]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i - 1][j] + pre[i][j - 1] - pre[i - 1][j - 1] + mat[i][j]$$

- Por ejemplo, para $pre[3][4]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i-1][j] + pre[i][j-1] - pre[i-1][j-1] + mat[i][j]$$

- Por ejemplo, para $pre[3][4]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i - 1][j] + pre[i][j - 1] - pre[i - 1][j - 1] + mat[i][j]$$

- Por ejemplo, para $pre[3][4]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i - 1][j] + pre[i][j - 1] - pre[i - 1][j - 1] + mat[i][j]$$

- Por ejemplo, para $pre[3][4]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

La formula es:

$$pre[i][j] = pre[i - 1][j] + pre[i][j - 1] - pre[i - 1][j - 1] + mat[i][j]$$

- Por ejemplo, para $pre[3][4]$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

```
int n, m;
int mat[n][m];
int pre[n][m];

for(int i = 0; i < n; i++) {
    for (int j = 0; j < m; j++) {
        pre[i][j] = 0;
        if (i > 0) pre[i][j] += pre[i-1][j];
        if (j > 0) pre[i][j] += pre[i][j-1];
        if (i > 0 && j > 0) pre[i][j] -= pre[i-1][j-1];
        pre[i][j] += mat[i][j];
    }
}
```

En matrices 2D

- Queremos obtener la suma de valores entre las filas x_1 y x_2 , y las columnas y_1 y y_2
- Por ejemplo $x_1 = 2, x_2 = 3, y_1 = 3, y_2 = 4$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener la suma de valores entre las filas a y A , y las columnas b y B
- Por ejemplo $a = 2, A = 3, b = 3, B = 4$
- Aplicamos la formula
$$pre[A][B] - pre[a - 1][B] - pre[A][b - 1] + pre[a - 1][b - 1]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener la suma de valores entre las filas a y A , y las columnas b y B
- Por ejemplo $a = 2, A = 3, b = 3, B = 4$
- Aplicamos la formula
$$pre[A][B] - pre[a - 1][B] - pre[A][b - 1] + pre[a - 1][b - 1]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener la suma de valores entre las filas a y A , y las columnas b y B
- Por ejemplo $a = 2, A = 3, b = 3, B = 4$
- Aplicamos la formula
$$pre[A][B] - pre[a - 1][B] - pre[A][b - 1] + pre[a - 1][b - 1]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener la suma de valores entre las filas a y A , y las columnas b y B
- Por ejemplo $a = 2, A = 3, b = 3, B = 4$
- Aplicamos la formula
$$pre[A][B] - pre[a - 1][B] - pre[A][b - 1] + pre[a - 1][b - 1]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2

En matrices 2D

- Queremos obtener la suma de valores entre las filas a y A , y las columnas b y B
- Por ejemplo $a = 2, A = 3, b = 3, B = 4$
- Aplicamos la formula
$$pre[A][B] - pre[a - 1][B] - pre[A][b - 1] + pre[a - 1][b - 1]$$

	1	2	3	4	5
1	1	2	1	3	2
2	3	2	2	1	1
3	1	2	1	3	2
4	1	1	2	1	2