Conjunto Capicúa

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Descripción**

Dado un conjunto de 1<=N<=1000 dígitos, determinar si se trata de un conjunto capicúa. Un conjunto capicúa se lee igual de izquierda a derecha que de derecha a izquierda, por ejemplo: 1 3 4 3 1, es un conjunto capicúa, mientras que 1 3 4 5 1 no lo es.

**Entrada**

En la primera línea el número de dígitos en el conjunto. En la segunda línea los N dígitos separados por espacios.

**Salida**

La palabra SI, si es un conjunto capicúa, o NO si no lo es.

**Ejemplo**

|  |  |
| --- | --- |
| Entrada  5  1 3 4 3 1 | Salida  SI |

*Fuente: lacj*

Problema subido por: [lacj20](https://omegaup.com/profile/lacj20/)

| **ID** | **Status** | **Porcentaje** | **Penalty** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/Conjunto-Capicua#problems/new-run) | | | | | | | | |
| 332714c8 | Respuesta correcta | 100.00% | 0 | cpp | 1.82 | 0.00 |  |  |
| 2a66c458 | Respuesta correcta | 100.00% | 0 | cpp | 3.04 | 0.02 |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/Conjunto-Capicua#problems>

#include <stdio.h>

void esCapicua(int arr[], int n){

int i = 0, j = n - 1;

while(i< j) {

if(arr[i] != arr[j] ) {

printf("NO");

return;

}

i++;

j--;

}

printf("SI");

}

int main() {

int N;

scanf("%d",&N);

int arr[N];

for(int i = 0; i < N; i++) {

scanf("%d",&arr[i]);

}

esCapicua(arr, N);

return 0;

}