Carretes

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

**Descripción**

En un tipo de caja registradora antigua, para mostrar los precios, se utilizaba un carrete con los dígitos del 0 al 9 escritos alrededor, para representar cada una de las cifras. este carrete podía girar, hacia delante o hacia atrás, avanzando de uno en uno los dígitos hasta legar al numero deseado y girando cada uno hacia el sentido que requiera menor numero de pasos Tu deseas conocer cuantos de estos pasos realiza la maquina para cambiar de un precio a otro.

**Problema**

Debes realizar un programa que, conociendo la cantidad de carretes que tiene la maquina, el precio inicial y el precio final, calcule el numero de pasos que realizara el mecanismo para llegar de uno a otro.

**Entrada**

Tu programa deberá leer del teclado los siguientes datos: en la primera línea el número de carretes N, que puede ser desde 2 hasta 200. En la segunda línea, el precio inicial como una cadena de N caracteres y en la tercera línea el precio final. Como otra cadena de caracteres.

**Salida**

Tu programa debe escribir en pantala el número total de pasos que realizará la máquina para llegar del precio inicial al precio final

**Ejemplos**

| **Entrada** | **Salida** |
| --- | --- |
| 4  5732  1598 | 14 |

**Consideraciones**

Tu programa se evaluará con varios casos de prueba.

**TIPS**

Recuerda que un digito char es diferente de un digito int ya que los char representan codigos ASCII Por tal motivo si quieres convertir un digito char a int debes restarle el valor ascii (char) '0'

El código ASCII lo puedes ver en http://ascii.cl/es/ y el código ascci es el mismo que usas cuando oprimes ALT+ número (del teclado númerico) de tu computadora. por ejemplo alt-64 es arroba y alt 92 es la diagonal invertida y alt-48 es el cero y alt-49 es e número 1 y alt-50 es el número 2

Cuando hacemos: char valor='a'; // lo podemos leer como "valor" es igual al codigo ascii de la letra a ( 'a' )

A continuación vamos a convertir 3 varables char con valores ASCII en sus valores enteros

int dig1=0, dig2=0,dig3=0; char car1='2'; // e' '2' equivale al 50 en código ASCII char car2='1'; // el '1' equivale al 49 en codigo ASCII char car3='3'; // el '3' equivale al 49 en codigo ASCII

// El ascii de cero ( '0' ) es igual a 48' // asi que si al ascii '1' (49) le restamos ascii '0' (48) tendemos 1 binario o int // si al ascii '0' (48) le restamos ascii '0' (48) tendremos 0 binario o int // si al ascii '3' (51) le restamos ascii '0'(48) tendremos 3 binario o int

// De tal manera que: dig1= car1-'0'; /// dig1 almacenara el valor entero (int o binario) 2 dig2= car2-'0'; /// dig3 almacenara el valor entero (int o binario) 1 dig3= car3-'3'; /// dig3 almacenara el valor entero (int o binario) 3

// para convertir los caracteres car1, car2 y car3 a un numero de 3 digitos // debes multiplicar por 10 para desplazar una potencia y sumar el digito como entero

int numero=0; numero=numero*10 + car1-'0' // numero = 2 numero=numero*10 + car2-'0' // numero = 21 numero=numero\*10 + car3-'0' // numero = 213

*Fuente: OMI Xalapa*

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

| **ID** | **Status** | **Porcentaje** | **Penalty** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/Carretes#problems/new-run) | | | | | | | | |
| a3ca6fdb | Respuesta correcta | 100.00% | 0 | cpp | 2.97 | 0.00 |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/Carretes#problems>

#include <iostream>

#include <stdio.h>

using namespace std;

int pasos(int inicial, int final)

{

int pasos\_derecha = 0;

int indice = inicial;

while (indice != final)

{

pasos\_derecha++;

indice++;

if (indice > 9)

{

indice = 0;

}

}

int pasos\_izquierda = 0;

indice = inicial;

while (indice != final)

{

pasos\_izquierda++;

indice--;

if (indice < 0)

{

indice = 9;

}

}

return std::min(pasos\_derecha, pasos\_izquierda);

}

int main() {

int N;

scanf("%d", &N);

string inicial; // = "5732"; // Console.ReadLine();

string final; // = "1598"; // Console.ReadLine();

cin >> inicial;

cin >> final;

int sum\_pasos = 0;

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

{

sum\_pasos += pasos((int)inicial[i]-'0', (int)final[i]-'0');

}

printf("%d\n", sum\_pasos);

system("pause");

return 0;

}