Camp. de Otoño 2025 - Apertura (intermedio)

06 de noviembre, 2025

Libro de soluciones

Este documento contiene las soluciones esperadas para los 7 problemas usados en el concurso de apertura de nivel intermedio.

Las siguientes personas apoyaron desarrollando el set de problemas ya sea creando o mejorando enunciados, soluciones, casos de prueba, verificadores de entradas y salidas:

Ignacio Benitez Salvador

Brandom Galder Hernández

Nicole Abigail Chow-Flores

Eduardo Soria

Jazmin Virgilio

Ashley Torres

Ulises Refugio

Christopher Gael Contla

Carlos Alberto Lara

Sandro Martínez

Ángel David Franco

D13O25. ¡Queremos calavera!

Por: Ashley Torres

Pista: Observa cuidadosamente los dulces que eligieron Fido y Bartolomeo. ¿Son exactamente las mismas letras en el mismo orden?

Solución:
El problema pide verificar si las dos cadenas son idénticas. Si el dulce F que eligió Fido es exactamente igual al B que eligió Bartolomeo, se imprime "SI"; en caso contrario, se imprime "NO".

Por ejemplo, si ambos eligieron "chocolate", son iguales y la salida es "SI". Si uno eligió "gomitas" y el otro "caramelo", no son iguales, por lo tanto la salida es "NO".

Complejidad: La operación de comparar dos strings es de complejidad $O(N)$, esto es debido a que se debe comparar cada carácter de cada string para encontrar una desigualdad. Si ambos strings son de longitud N entonces la complejidad de comparar strings es $O(N)$.

D14O25. La metamorfosis de Gmail (el perrito)

Por: Hector Ulises Refugio Becerril

Pista 1: Un número X es divisible entre Y si el residuo de su división es $X \,\%\, Y = 0$
Pista 2: Un número X es par si es divisible entre 2.

Solución:
La solución consiste en aplicar operaciones aritméticas básicas y el uso de condiciones.

- En el primer punto se debe comprobar si un número X es divisible entre Y.
- En el segundo punto basta con emplear los operadores de comparación para saber si X es mayor, menor o igual que Y.
- Finalmente, en el tercer punto se solicita determinar la paridad del residuo de X entre Y. Para ello obtenemos el residuo de la división mediante el operador módulo y después se verifica si dicho valor es par o impar.

Complejidad: Todas las operaciones utilizadas (comparaciones y operaciones aritméticas básicas) se realizan en tiempo constante $O(1)$, por lo que la complejidad total es $O(1)$.

C11O25. La suma otoñal

Por: Nicole Abigail Chow-Flores

Pista 1: La acumulación de hojas hasta un día d, es igual a $\sum_{i=1}^{d} X_i$.
Pista 2: La acumulación de hojas hasta un día d, también es igual a $\left[\sum_{i=1}^{d-1} X_i\right] + X_d$.

Solución:
Para cada día i desde 1 hasta N, recibirás un valor X_i, el cual indica cuántas hojas se recolectaron en el i-*ésimo* día.

Dado que la acumulación de hojas hasta un día d, también es igual a $\left[\sum_{i=1}^{d-1} X_i\right] + X_d$, debemos iterar sobre cada día d y obtener esa acumulación.

El primer índice d tal que $\left[\sum_{i=1}^{d-1} X_i\right] + X_d \geq K$ es la respuesta. Si no hay alguna d que cumpla la condición anterior, deberás imprimir -1.

Complejidad: Como iterar sobre N elementos tiene complejidad $O(N)$, y hacer una suma tiene complejidad $O(1)$, a complejidad es $O(N \cdot 1) = O(N)$.

C12O25. El frío se acerca oíste viejo

Por: Brandom Galder Hernández

Pista 1: Los números de la sumatoria son consecutivos.
Pista 2: La sumatoria de números consecutivos desde 1 hasta N se calcula en $O(1)$ empleando la Sumatoria de Gauss.

Solución:
Por ejemplo, cuando $L = 1$ y $R = 5$, el resultado es: $1+2+3+4+5 = 15$. Podemos observar que el resultado se obtiene haciendo la sumatoria de números consecutivos desde 1 hasta 5. Formular la Sumatoria de Gauss para dicho ejemplo se vería:

$$\sum_{i=1}^{N} i = \frac{N*(N+1)}{2} = \frac{5*(5+1)}{2} = 15$$

Para este problema, la fórmula funciona cuando $L = 1$ porque la Sumatoria de Gauss suma desde 1 hasta N, pero ¿qué hacemos cuando $L > 1$? Si por ejemplo $L = 4$ y $R = 5$, el resultado es $4 + 5 = 9$. Está claro que no sumamos $1 + 2 + 3 = 6$, sin embargo, efectuar la Sumatoria de Gauss hace la suma $1 + 2 + 3 + 4 + 5 = 15$. Lo que podemos hacer es restar los números que no queremos en nuestra suma, es decir, $[1+2+3+4+5]-[1+2+3] = 9$, lo que equivale a:

$$\left[\sum_{i=1}^{R} i\right] - \left[\sum_{j=1}^{L-1} j\right] = \left[\frac{R*(R+1)}{2}\right] - \left[\frac{(L-1)*((L-1)+1)}{2}\right] =$$

$$\left[\frac{5*(5+1)}{2}\right] - \left[\frac{3*(3+1)}{2}\right] = [1+2+3+4+5]-[1+2+3] = 15-6 = 9$$

Esta operación funciona para cualquier L, entonces la respuesta es imprimir el resultado de:

$$\left[\sum_{i=1}^{R} i\right] - \left[\sum_{j=1}^{L-1} j\right]$$

Complejidad: Calcular ambas sumatorias (hasta $L-1$ y hasta R) se realiza en $O(2)$. Dado que son las únicas operaciones, la complejidad final es prácticamente $O(1)$.

C13O25. Hojitas en los árboles

Por: Jazmin Virgilio Escamilla

Pista 1: No importa el orden inicial de las hojas, ya que puedes reordenarlas libremente.
Pista 2: Cada hoja debe tener un tamaño único.

Solución:
Dado el arreglo $A = \{A_1, A_2, \ldots, A_N\}$, se debe determinar si es posible reordenarlo de manera que quede estrictamente creciente. Para verificarlo, simplemente se ordena el arreglo A; tras el ordenamiento, basta con revisar que para todo $1 \leq i < N$ se cumple $A_i < A_{i+1}$.

Si para por lo menos una i se tiene que $A_i = A_{i+1}$, entonces existen elementos repetidos y no es posible obtener una secuencia estrictamente creciente. Por lo tanto, el arreglo es estrictamente creciente cuando **TODOS** los elementos de A son distintos.

Complejidad: El ordenamiento mediante `sort` en C++ tiene complejidad $O(N \log N)$; posteriormente, se recorre el arreglo una sola vez, realizando $N - 1$ comparaciones, lo cual tiene complejidad $O(N - 1) \approx O(N)$. La complejidad total es $O(N + [N \cdot\ log(N)])$.

C14O25. Antivirus Pirata

Por: Eduardo Soria

Pista: Se puede usar un arreglo de frecuencias de tamaño N para contabilizar los ataques del virus.

Solución:
La forma más sencilla de simular el comportamiento del virus es usando un arreglo de frecuencias, donde la *i-ésima* posición representa el número de ataques que recibió el *i-ésimo* archivo.

El proceso es el mismo vuelta por vuelta. En la vuelta j el virus avanza dando saltos de tamaño j, por lo que debes aumentar en 1 la frecuencia de los archivos j, $2j$, $3j$, ..., atacando todos los archivos $x \cdot j$ tal que $x \cdot j \leq N$. Este proceso se repite desde $j = 1$ hasta $j = N$.

Al finalizar, cada elemento del arreglo de frecuencias tendrá almecenada las veces en las que fueron atacados los archivos, por lo que será necesario recorrer el arreglo y contar cuántos elementos son impares.

Complejidad:
Atacar los archivos en la *j-ésima* iteración tiene complejidad $O(\frac{N}{j})$. Realizar esto N veces, ejecuta

$$\sum_{j\,=\,1}^{N} \frac{N}{j} = \frac{N}{1} + \frac{N}{2} + \ ... \ + \frac{N}{N} = N \ \left(\frac{1}{1} + \frac{1}{2} + \ ... \ + \frac{1}{N} \right)$$

operaciones. La sumatoria representa una serie armónica, la cual crece aproximadamente tan rápido como $ln(N)$, por lo tanto, la complejidad equivale a $O(N \cdot \ log(N))$.

Recorrer el arreglo tiene complejidad $O(N)$. La complejidad total es $O(N + [N \cdot \ log(N)])$

C15O25. Piénsale tantito, Maullin

Por: Ignacio Benítez Salvador

Pista 1: Obtén el valor máximo y el mínimo.
Pista 2: Si ordenas los lugares, puedes tener un mejor panorama.

Solución:
Todos los puntos están entre el mínimo L y el máximo R, forman un rango $[L, R]$, donde L es el menor valor y R el mayor.

Para visitar todos empezando en un punto B, lo óptimo es:

- Ir primero al extremo (L o R) más cercano a B, para evitar recorrer tramo innecesario. Esto se obtiene con: $\min(|B - L|, |R - B|)$.
- Luego, desde ese extremo, solo es recorrer hasta el otro: $(R - L)$. Para entender por qué solo importa la diferencia entre los extremos, imagina los puntos en orden:

  $$1, 2, 3, 4, 5$$

  Si sumamos las distancias consecutivas:

  $$|1 - 2| + |2 - 3| + |3 - 4| + |4 - 5|$$

  Esto equivale a:

  $$(2 - 1) + (3 - 2) + (4 - 3) + (5 - 4)$$

  Al expandir y simplificar, todos los valores intermedios se cancelan:

  $$-1 + 2 - 2 + 3 - 3 + 4 - 4 + 5 = (5 - 1)$$

  Sin importar cuántos puntos haya, recorrerlos todos una vez desde un extremo hasta el otro siempre suma exactamente $(R - L)$.

Por lo tanto, la fórmula final para cada consulta es:

$$\text{ans} = (R - L) + \min(|B - L|, |R - B|).$$

Complejidad: Obtener L y R es $O(N)$. Cada consulta se responde en $O(1)$, dando una complejidad total de $O(N + Q)$.