Ejercicio 5

Encontrar el algoritmo de búsqueda binaria para encontrar un elemento K en una lista de elementos $X_1, X_2, \dots, X_n$ previamente clasificados en orden ascendente.

El array o vector X se supone ordenado en orden creciente si los datos son numéricos, o alfabéticamente si son caracteres. Las variables BAJO, CENTRAL, ALTO indican los límites inferior, central y superior del intervalo de búsqueda.

Análisis:

Extracción y Naturaleza de los Datos

Matemáticamente, definimos nuestra lista de datos X no como un simple montón de números, sino como un **Conjunto Ordenado Finito**.

- Sea $X = \{x_0, x_1, x_2, \dots, x_{n-1}\}$.
- **Condición estricta de orden:** Para que este algoritmo funcione, se debe cumplir que $\forall i, j \in \mathbb{Z} : i < j \Rightarrow x_i \leq x_j$. (Si el índice $\boldsymbol{i}$ es menor que $\boldsymbol{j}$**,** el valor en x_i es menor o igual al de x_j).

El Espacio de Búsqueda (Intervalo Cerrado)

El algoritmo no busca en todo el array a la vez. Define un **intervalo de búsqueda** activo.

- Definimos el intervalo actual como $[L, R]$ (correspondiente a $\boldsymbol{BAJO}$ y $\boldsymbol{ALTO}$).
- Inicialmente, el intervalo es completo: $[0, N-1]$**.**
- La longitud del intervalo es $Longitud = R - L + 1$.

El Cálculo del Punto Medio ("CENTRAL")

La operación clave en la imagen es:

$$CENTRAL \leftarrow ent((BAJO + ALTO) / 2)$$

Matemáticamente, esto es el **promedio aritmético** truncado.

1. **Suma:** (BAJO + ALTO) nos da la suma de los índices extremos.
2. **División:** Al dividir por 2, buscamos el centro geométrico.
3. **Función Suelo (Floor):** La parte $\boldsymbol{ent}()$ o la división entera en C++ aplica la función suelo $\lfloor \cdot \rfloor$
   - Fórmula real: $M = \left\lfloor \frac{L+R}{2} \right\rfloor$
   - *Ejemplo:* Si $\boldsymbol{BAJO = 0\ y\ ALTO = 9}$ (10 elementos), M = $\lfloor 4.5 \rfloor = 4$. El elemento central es el índice 4.

La Lógica de las Desigualdades (Reducción Logarítmica)

Aquí es donde ocurre la "magia" matemática. Comparamos el valor buscado K con el valor central x_M.

Caso 1: $K < x_M$ (La izquierda)

- **Axioma:** Como el array está ordenado, si K es menor que el valor del centro, **necesariamente** K debe ser menor que cualquier valor a la derecha del centro.
- **Deducción matemática:** $K \notin [M, R]$.
- **Operación:** Descartamos todo el subconjunto derecho. El nuevo límite superior **(ALTO)** se convierte en $M - 1$.
- Nuevo intervalo: $[L, M - 1]$.

Caso 2: $K > x_M$ (La derecha)

- **Axioma:** Si K es mayor que el centro, es imposible que esté a la izquierda.
- **Deducción matemática:** $K \notin [L, M]$.
- **Operación:** Descartamos todo el subconjunto izquierdo. El nuevo límite inferior **(BAJO)** se convierte en $M + 1$.
- Nuevo intervalo: $[M + 1, R]$.