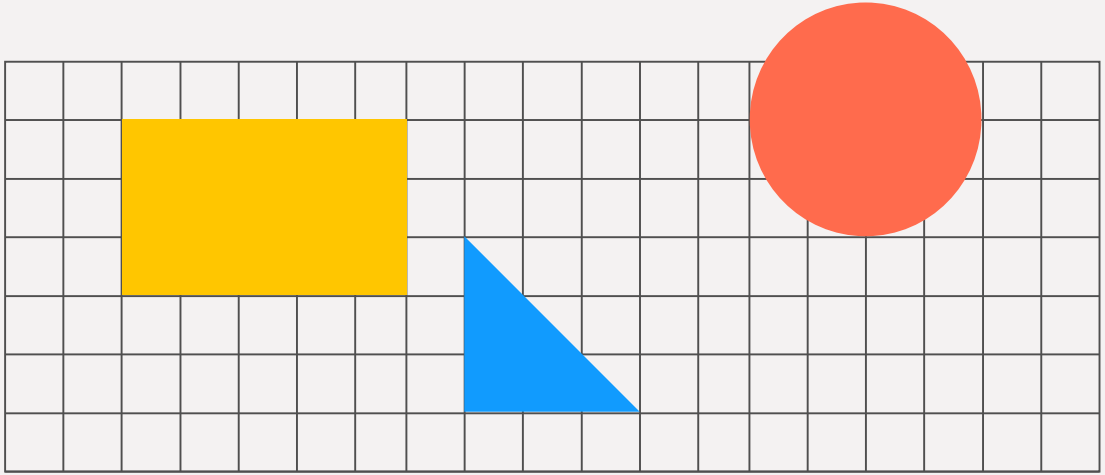
Unidad 4
Búsqueda Binaria



Búsqueda Binaria

La búsqueda binaria es un algoritmo eficiente para encontrar un elemento en un **arreglo ordenado**. A diferencia de la búsqueda lineal (que revisa uno por uno), la binaria divide el arreglo por la mitad en cada paso, descartando la mitad en la que el elemento no puede estar.

```c
int main() {
    int arreglo[] = {3, 7, 15, 20, 25, 31, 40}; // Arreglo ORDENADO
    int n = sizeof(arreglo) / sizeof(arreglo[0]);
    int buscar, inicio = 0, fin = n - 1, medio;
    int encontrado = 0;
    printf("Ingrese el numero que desea buscar: ");
    scanf("%d", &buscar);
    while (inicio <= fin) {
        medio = (inicio + fin) / 2;

        if (arreglo[medio] == buscar) {
            printf("Numero encontrado en la posicion %d.\n", medio);
            encontrado = 1;
            break;
        } else if (buscar < arreglo[medio]) {
            fin = medio - 1;
        } else {
            inicio = medio + 1;
        }
    }
    if (!encontrado) {
        printf("El numero no se encuentra en el arreglo.\n");
    }
    return 0;
}
```

Requisitos

- El arreglo debe estar ordenado (ascendente o descendente).
- Es más rápida que la búsqueda lineal, especialmente en arreglos grandes.

¿Cómo funciona?

- Se comparan los extremos: izquierda (`inicio`) y derecha (`fin`).
- Se calcula el **índice del medio**:
  `medio = (inicio + fin) / 2`
- Se compara el valor en esa posición con el valor buscado:
  - Si son iguales → **¡Listo! Elemento encontrado.**
  - Si el buscado es menor → Buscar en la **mitad izquierda**.
  - Si es mayor → Buscar en la **mitad derecha**.
- Se repite el proceso hasta encontrarlo o que no queden elementos.

Ventajas de la búsqueda binaria

Ventaja	Detalle
Muy eficiente	O(log n) tiempo
Rápida en listas grandes	Ideal cuando hay muchos elementos
Menos comparaciones	Que la búsqueda lineal

- NO FUNCIONA SI EL ARREGLO NO ESTÁ ORDENADO.