



ICC311

Estructuras de Datos

Semestre I, 2020

Profesor: Pablo Valenzuela

Semana 01 - Parte 02_a

Limitaciones en el uso de Arreglos

1. Organización de datos
2. Limitaciones en el uso de arreglos



Organización de Datos

- Agregar un número a una colección de datos
 - ◆ Eventualmente en cierta posición
- Eliminar un número desde una colección de datos
 - ◆ Eventualmente desde cierta posición
- Determinar si en una colección de datos existe cierto número
- Recuperar un número desde cierta ubicación
- Recuperar números en cierto orden
- Recuperar números en el orden en que fueron insertados

nums[0]	nums[1]	nums[2]	nums[3]
28	14	17	30

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 28;
```

```
int nums[] = { 28,14,17,30 };
```

**¿ Qué tipo de limitaciones existen
en el uso de Arreglos ?**

Limitación #1:

Se debe conocer el tamaño del arreglo anticipadamente

nums[0]	nums[1]	nums[2]	nums[3]
28	14	17	30

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;  
  
. . .  
  
nums[4] = 16;
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]
28	14	17	30	X

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;  
  
. . .  
  
nums[4] = 16;
```


nums[0]	nums[1]	nums[2]	nums[3]	nums[4]
28	14	17	30	X

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;
```

. . .

```
nums[4] = 16;
```

Index Out of Bound Exception

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	?	?	?	?

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;
```

Limitación #2:

**No es posible insertar (o eliminar) un elemento
delante de otros de manera sencilla**

				cont = 4			
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	?	?	?	?

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;
```

```
int cont = 4; // número de elementos en el arreglo
```

				cont = 4			
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	16	?	?	?

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;  
  
int cont = 4; // número de elementos en el arreglo  
  
nums[cont] = 16; // se agrega al final
```

					cont = 4		
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	16	?	?	?

```
int nums[] = new int[4];  
nums[0] = 28;  
nums[1] = 14;  
nums[2] = 17;  
nums[3] = 30;  
  
int cont = 4; // número de elementos en el arreglo  
  
nums[cont] = 16; // se agrega al final  
count++;
```

					cont = 4		
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	16	41	?	?

```
int nums[] = new int[4];
nums[0] = 28;
nums[1] = 14;
nums[2] = 17;
nums[3] = 30;

int cont = 4; // número de elementos en el arreglo

nums[cont] = 16;
count++;
nums[cont] = 41; // se agrega al final
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	16	41	41	?

22

```
nums[6] = nums[5];
```


nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	16	16	41	?

22

```
nums[6] = nums[5];  
nums[5] = nums[4];
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	30	30	16	41	?

22

```
nums[6] = nums[5];  
nums[5] = nums[4];  
nums[4] = nums[3];
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	17	17	30	16	41	?

22

```
nums[6] = nums[5];  
nums[5] = nums[4];  
nums[4] = nums[3];  
nums[3] = nums[2];
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	14	14	17	30	16	41	?

22

```
nums[6] = nums[5];  
nums[5] = nums[4];  
nums[4] = nums[3];  
nums[3] = nums[2];  
nums[2] = nums[1];
```

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	?

22

```
nums[6] = nums[5];  
nums[5] = nums[4];  
nums[4] = nums[3];  
nums[3] = nums[2];  
nums[2] = nums[1];  
nums[1] = 22;
```

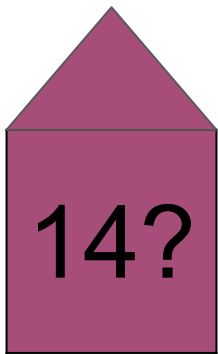
Limitación #3:

Algunas operaciones pueden ser muy lentas

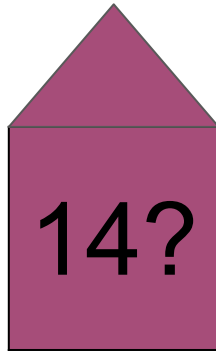
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11

14

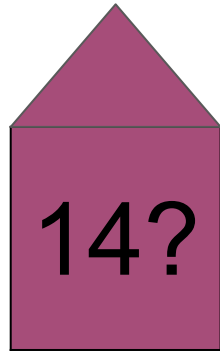
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



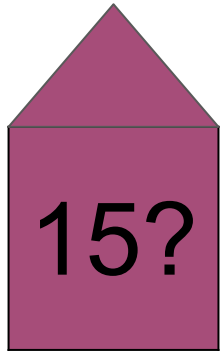
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



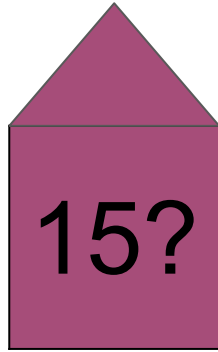
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11

15

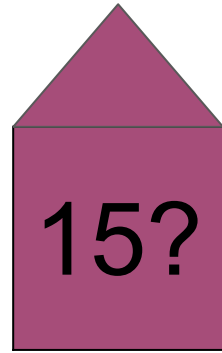
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



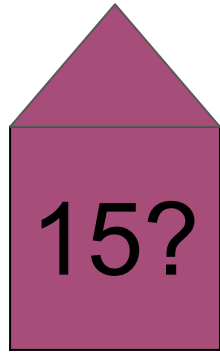
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



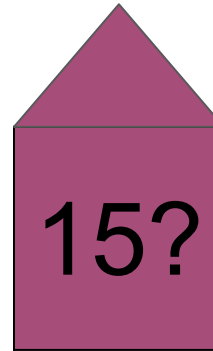
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



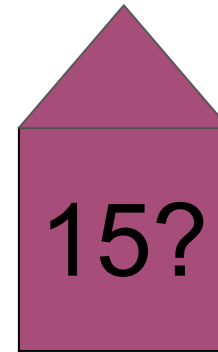
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



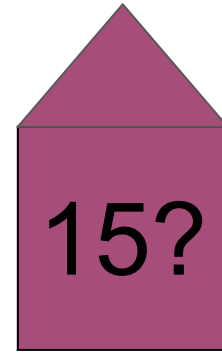
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



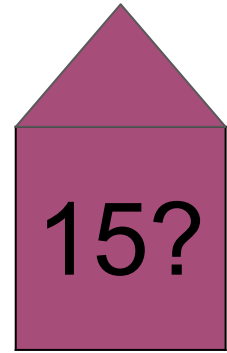
nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11



nums[0]	nums[1]	nums[2]	nums[3]	nums[4]	nums[5]	nums[6]	nums[7]
28	22	14	17	30	16	41	11

```
public boolean contiene ( int[] nums, int objetivo ) {  
    for ( int i = 0; i < nums.length; i++ ) {  
        if ( nums[i] == objetivo ) {  
            return true; // se encontró el número  
        }  
    }  
    return false; // no se encontró el número  
}
```

Limitaciones en el uso de Arreglos

1. Se debe conocer el tamaño del arreglo anticipadamente
 - ◆ Muy pequeño : no se puede agregar elementos
 - ◆ Muy grande : se pierde espacio
2. No es posible insertar (o eliminar) un elemento delante de otros de manera sencilla
 - ◆ Se debe mover todos los elementos, y se debe asumir que se cuenta con suficiente espacio
3. Algunas operaciones pueden ser muy lentas



ICC311

Estructuras de Datos

Semestre I, 2020

Profesor: Pablo Valenzuela