

Recursividad

24130836

1. Invertir (string cad)

string res = "";

for (int i = cad.length() - 1; i >= 0; i--)

{  
    resultado += cadena.charAt(i);

}

a-l-o-h  
0 1 2 3

← h-o-l-a  
0 1 2 3

0, 1, 2, ...

return resultado

Recursive (indice) → cadena.length - 1  
hola → 4 - 1 = 3

if (indice < 0) {  
    return "";

} else {

return cadena.charAt(indice) + invertir(cadena, indice - 1);

## Suma de Dígitos

$$\text{suma} += n \% 10 = 5 \quad 564$$

$$n = n / 10 = 64 \quad 5 + 6 + 4 = 15$$

while  $n > 0$  ↑

if  $(n == 0)$   
return 0

else

return  $(n \% 10) + \text{sumaDígitosR}(n / 10);$

$$5 + 6 + 4 = 15 \leftarrow 564$$

↻



```

Contar Vocales (string Cadena, indice)
if indice >= Cadena.length()
    return 0
else
    char c = Character.toLowerCase(Cadena.charAt(indice))
    int vocal = 0;
    if (c == 'a' || c == 'e' || c == 'i' || c == 'o')
        vocal = 1;
    return vocal + contarVocales(Cadena, indice + 1)

```

3 (indice >= ...) Programación  
 ↑  
 0, 1, 2, 3, ...

```
buscarNumeroR(int[] arreglo, int numero, indice)
```

```
if (indice >= arreglo.length
```

```
return "No encontrado"
```

```
else
```

```
if (arreglo[indice] == numero)
```

```
return "Encontrado";
```

```
else
```

```
return buscarNumeroR(arreglo, numero,  
indice+1)
```

0	1	2	3	4	5	
{	3	, 7	, 12	, 19	, 5	}

19





↑ boolean

PalindromoC(String palabra, int izquierda, int derecha)

palabra = palabra.toLowerCase();

if (izquierda >= derecha)

return true

else

if (palabra.charAt(izquierda) != ... (derecha))

return false;

else

return PalindromoC(palabra, izquierda+1,  
derecha-1)

izquierda      derecha

0

palabra.length()-1