

# RECURRENCIAS



- Recurrencias

# Recurrencias

---

**Problema:** Dado un arreglo ordenado encontrar el valor b

|    |    |   |   |   |    |    |    |    |    |
|----|----|---|---|---|----|----|----|----|----|
| -3 | -2 | 0 | 3 | 7 | 11 | 14 | 22 | 26 | 34 |
|----|----|---|---|---|----|----|----|----|----|

## Programa 1:

```
public void buscar(int b){  
    for(int i=0; i<n; i=i+1){  
        if (datos[i]==b){  
            System.out.println("Encontrado");  
            break;  
        }  
    }  
}
```

## Programa 1:

```
public void buscar(int b){  
    for(int i=0; i<n; i=i+1){  
        if (datos[i]==b){  
            System.out.println("Encontrado");  
            break;  
        }  
    }  
}  
}
```

|    |    |   |   |   |    |    |    |    |    |
|----|----|---|---|---|----|----|----|----|----|
| 0  | 1  | 2 | 3 | 4 | 5  | 6  | 7  | 8  | 9  |
| -3 | -2 | 0 | 3 | 7 | 11 | 14 | 22 | 26 | 34 |

**buscar(26)**

## Programa 1:

```
public void buscar(int b){  
    for(int i=0; i<n; i=i+1){  
        if (datos[i]==b){  
            System.out.println("Encontrado");  
            break;  
        }  
    }  
}
```

En el **peor caso**, ¿cuántas comparaciones hará el programa para un arreglo de tamaño  $n$ ?

## Programa 2:

```
public void buscar(int i, int j, int b){  
    int medio = Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
        break;  
    }  
    if (datos[medio]<b)  
        buscar(medio,j,b);  
    if (datos[medio]>b)  
        buscar(i,medio,b);  
}
```

## Programa 2:

```
public void buscar(int i, int j, int b){  
    int medio = Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
        break;  
    }  
    if (datos[medio]<b)  
        buscar(medio,j,b);  
    if (datos[medio]>b)  
        buscar(i,medio,b);  
}
```

|    |    |   |   |   |    |    |    |    |    |
|----|----|---|---|---|----|----|----|----|----|
| 0  | 1  | 2 | 3 | 4 | 5  | 6  | 7  | 8  | 9  |
| -3 | -2 | 0 | 3 | 7 | 11 | 14 | 22 | 26 | 34 |

buscar(0,9,26)

## Programa 2:

```
public void buscar(int i, int j, int b){  
    int medio = Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
        break;  
    }  
    if (datos[medio]<b)  
        buscar(medio,j,b);  
    if (datos[medio]>b)  
        buscar(i,medio,b);  
}
```

En el peor caso, ¿cuántas comparaciones hará el programa para un arreglo de tamaño n?

# Recurrencias

## Programa 1:

```
public void buscar(int b){  
    for(int i=0; i<n; i=i+1){  
        if (datos[i]==b){  
            System.out.println("Encontrado");  
            break;  
        }  
    }  
}
```

## Programa 2:

```
public void buscar(int i, int j, int b){  
    int medio=Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
        break;  
    }  
    if (datos[medio]<b)  
        buscar(medio,j,b);  
    if (datos[medio]>b)  
        buscar(i,medio,b);  
}
```

# Recurrencias

## Programa 1:

```
public void buscar(int b){  
    for(int i=0; i<n; i=i+1){  
        if (datos[i]==b){  
            System.out.println("Encontrado");  
            break;  
        }  
    }  
}
```

## Programa 2:

```
public void buscar(int i, int j, int b){  
    int medio=Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
        break;  
    }  
    if (datos[medio]<b)  
        buscar(medio,j,b);  
    if (datos[medio]>b)  
        buscar(i,medio,b);  
}
```

$$T_1(n)=n$$

$$T_2(n)=\log_2 n$$

# Recurrencias

---

El tiempo de cómputo  $T$  de un algoritmo depende del tamaño de la entrada,

$$T(n) = f(n), \text{ donde } n \text{ es el tamaño de la entrada}$$

# Recurrencias

---

$$T_1(n) = n$$

$$T_2(n) = \log_2 n$$

Para  $n=4$ , se tiene:

$$T_1(4) = ?$$

$$T_2(4) = ?$$

# Recurrencias

---

$$T_1(n) = n$$

$$T_2(n) = \log_2 n$$

Para  $n=4$ , se tiene:

$$T_1(4) = 4$$

$$T_2(4) = \log_2 4 = 2$$

# Recurrencias

---

$$T_1(n) = n$$

$$T_2(n) = \log_2 n$$

Para  $n=64$ , se tiene:

$$T_1(64) = 64$$

$$T_2(64) = \log_2 64 = 6$$

# Recurrencias

---

$$T_1(n) = n$$

$$T_2(n) = \log_2 n$$

Para  $n=1024$ , se tiene:

$$T_1(1024) = 1024$$

$$T_2(1024) = \log_2 1024 = 10$$

# Recurrencias

---

$$T_1(n) = n$$

$$T_2(n) = \log_2 n$$

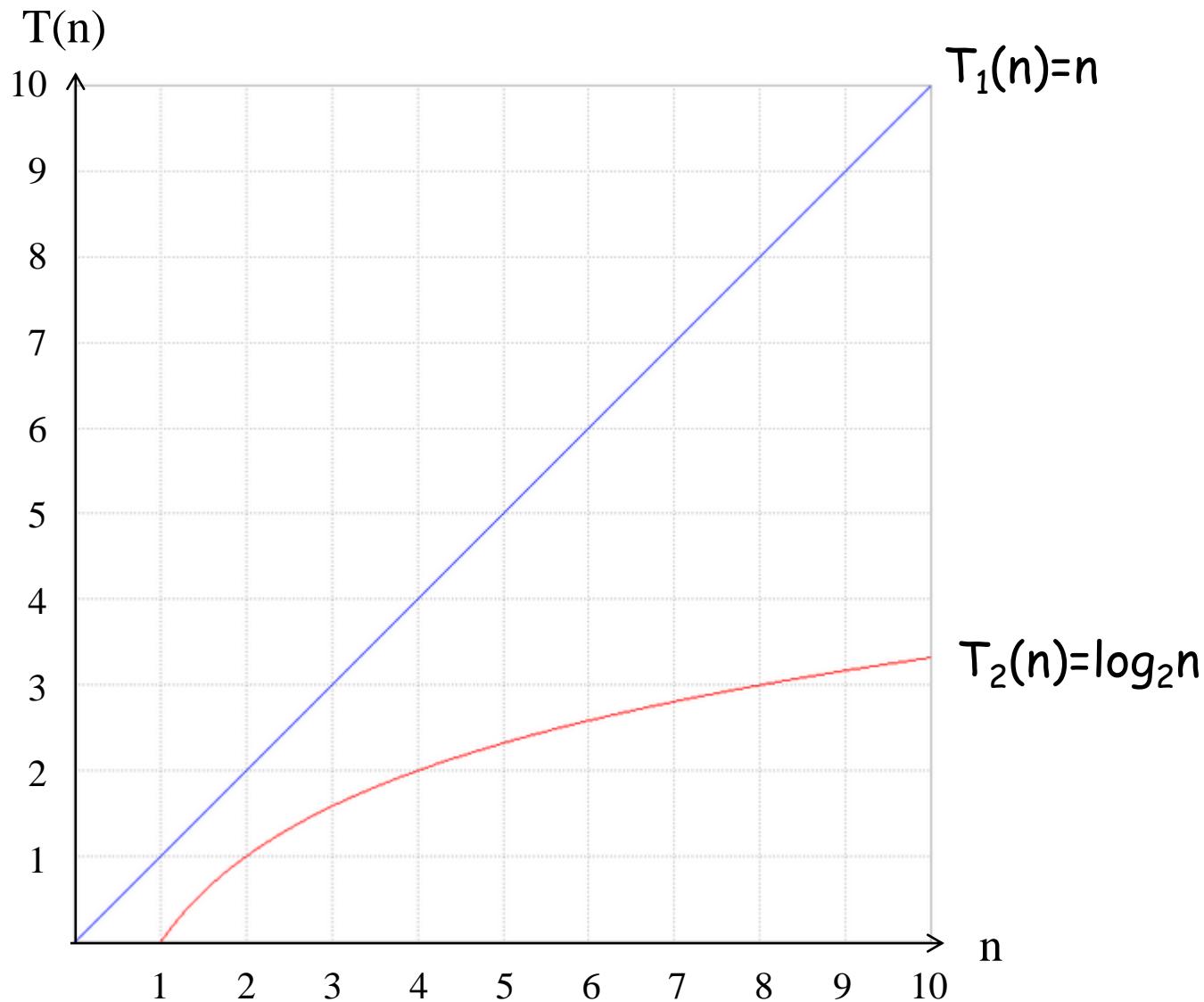
Para  $n=1024$ , se tiene:

$$T_1(1024) = 1024$$

$$T_2(1024) = \log_2 1024 = 10$$

|       | $n=4$ | $n=64$ | $n=1024$ |
|-------|-------|--------|----------|
| $T_1$ | 4     | 64     | 1024     |
| $T_2$ | 2     | 6      | 10       |

# Recurrencias



# Recurrencias

---

$T(n)$



$T_1(n)=n$

$T_2(n)=\log_2 n$

$n$

# Recurrencias

---

```
misterio(int n){
```

```
    if (n==1)
```

```
        return 1;
```

```
    else{
```

```
        int c=0;
```

```
        for (int i=2; i<n; i++){
```

```
            if (n%i==0)
```

```
                c++;
```

```
}
```

```
        if (c==0)
```

```
            return 1;
```

```
        else
```

```
            return 0;
```

```
}
```

# Recurrencias

---

```
misterio(int n){  
    if (n==1)  
        return 1;  
else{  
    int c=0;  
    for (int i = 2; i < n; i++){  
        if (n % i == 0)  
            c++;  
    }  
    if (c==0)  
        return 1;  
else  
    return 0;  
}
```

Calcular:

- **misterio(5)**
- **misterio(6)**
- **misterio(7)**

# Recurrencias

---

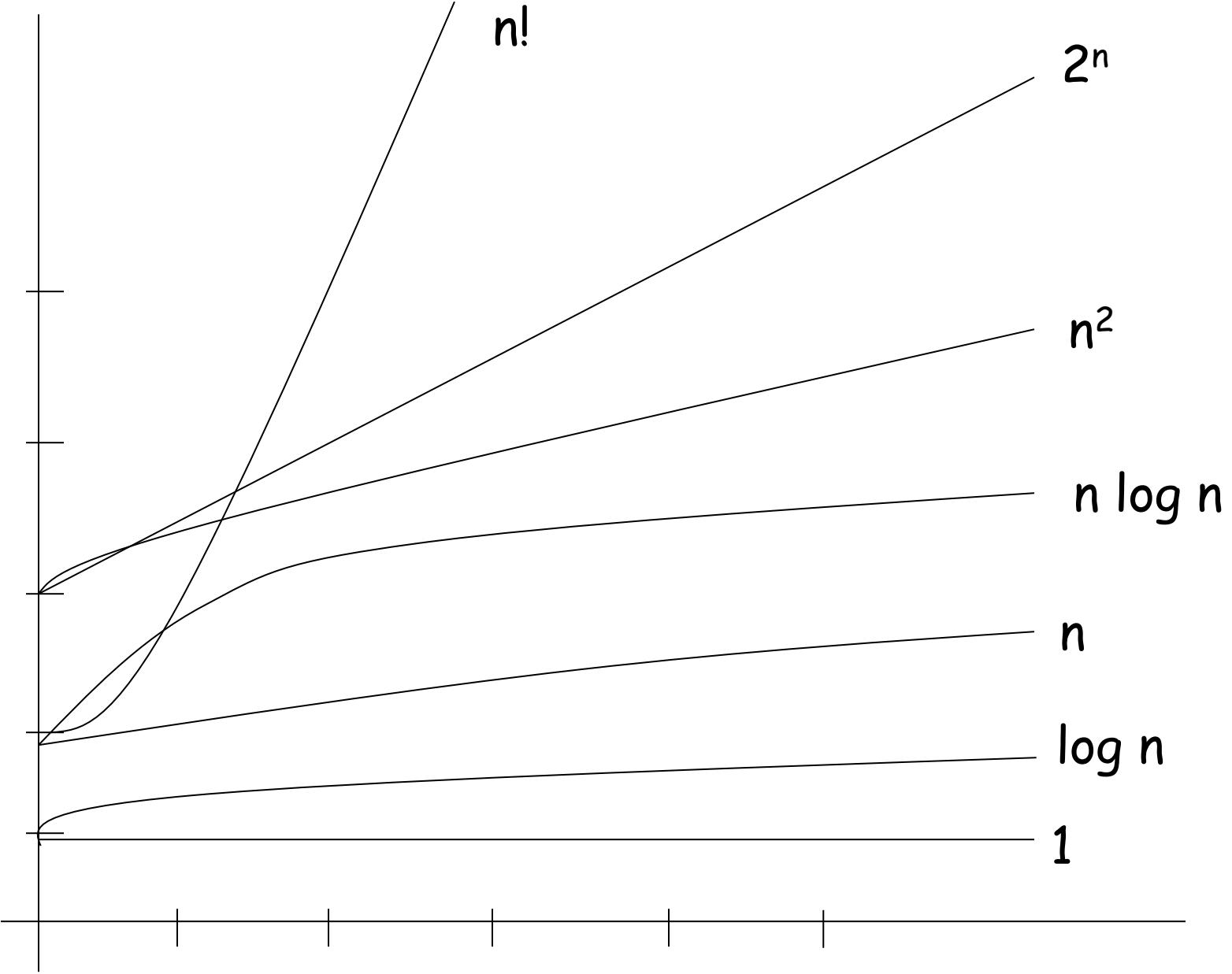
```
primo(int n){  
    if (n==1)  
        return 1;  
    else{  
        int c=0;  
        for (int i=2; i<n; i++){  
            if (n%i==0)  
                c++;  
        }  
        if (c==0)  
            return 1;  
        else  
            return 0;  
    }  
}
```

# Recurrencias

---

```
primo(int n){  
    if (n==1)  
        return 1;  
    else{  
        int c=0;  
        for (int i=2; i<n; i++){  
            if (n%i==0)  
                c++;  
        }  
        if (c==0)  
            return 1;  
        else  
            return 0;  
    }
```

*¿De qué depende la cantidad de operaciones básicas que realizará el algoritmo para un llamado específico?*



# Recurrencias

---

| Complejidad   | Terminología            |
|---------------|-------------------------|
| $O(1)$        | Complejidad constante   |
| $O(\log n)$   | Complejidad logarítmica |
| $O(n)$        | Complejidad lineal      |
| $O(n \log n)$ | Complejidad $n \log n$  |
| $O(n^b)$      | Complejidad polinómica  |
| $O(b^n)$      | Complejidad exponencial |
| $O(n!)$       | Complejidad factorial   |

# Recurrencias

---

## Recurrencias

Una **recurrencia** es una ecuación que define una secuencia recursiva, cada término de la secuencia se define como una función de los términos anteriores

## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?



## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$



## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = ?$$



## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = ?$$



## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, **¿cuánto dinero habrá en la cuenta al cabo de 30 años?**

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = ?$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$

**Recurrencia** que representa la cantidad de dinero en el banco

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$

**Recurrencia** que representa la cantidad de dinero en el banco

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$

**¿Cómo resolver la recurrencia?**



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 + P_0 \cdot 0.11 \quad P_1 = P_0 (1 + 0.11) \quad P_1 = P_0 (1.11)$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, **¿cuánto dinero habrá en la cuenta al cabo de 30 años?**

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_1 + P_1 \cdot 0.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_1 \cdot 1.11$$

$$P_3 = P_2 + P_2 \cdot 0.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_1 \cdot 1.11$$

$$P_3 = P_2 \cdot 1.11$$

...

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_1 \cdot 1.11$$

$$P_3 = P_2 \cdot 1.11$$

...

$$P_n = P_{n-1} \cdot 1.11$$



## Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$\mathbf{P_1} = P_0 \cdot 1.11$$

$$P_2 = P_1 \cdot 1.11 \quad P_2 = \mathbf{P_1} \cdot 1.11 \quad P_2 = (\mathbf{P_0 \cdot 1.11}) \cdot 1.11$$

$$P_3 = P_2 \cdot 1.11$$

...

$$P_n = P_{n-1} \cdot 1.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, **¿cuánto dinero habrá en la cuenta al cabo de 30 años?**

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = (P_0 \cdot 1.11) \cdot 1.11$$

$$P_3 = P_2 \cdot 1.11$$

...

$$P_n = P_{n-1} \cdot 1.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, **¿cuánto dinero habrá en la cuenta al cabo de 30 años?**

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = P_2 \cdot 1.11$$

...

$$P_n = P_{n-1} \cdot 1.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, **¿cuánto dinero habrá en la cuenta al cabo de 30 años?**

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = (P_0 \cdot (1.11)^2) \cdot 1.11$$

...

$$P_n = P_{n-1} \cdot 1.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = P_0 \cdot (1.11)^3$$

...

$$P_n = P_{n-1} \cdot 1.11$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = P_0 \cdot (1.11)^3$$

...

$$P_n = P_0 \cdot (1.11)^n$$



# Recurrencias

---

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = P_0 \cdot (1.11)^3$$

...

$$P_n = P_0 \cdot (1.11)^n$$

**La recurrencia**

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$

**se soluciona con la fórmula cerrada :**

$$P_n = P_0 \cdot (1.11)^n$$



# Recurrencias

Una persona deposita 10000 pesos en una cuenta bancaria que le proporciona un interés anual del 11%. Si los intereses se abonan a la misma cuenta, ¿cuánto dinero habrá en la cuenta al cabo de 30 años?

$$P_0 = 10000$$

$$P_1 = P_0 \cdot 1.11$$

$$P_2 = P_0 \cdot (1.11)^2$$

$$P_3 = P_0 \cdot (1.11)^3$$

...

$$P_n = P_0 \cdot (1.11)^n$$

**La recurrencia**

$$P_n = P_{n-1} + P_{n-1} \cdot 0.11$$

**se soluciona con la fórmula cerrada**

$$P_n = P_0 \cdot (1.11)^n$$

$$\mathbf{P_{30} = 10000 \cdot (1.11)^{30}}$$

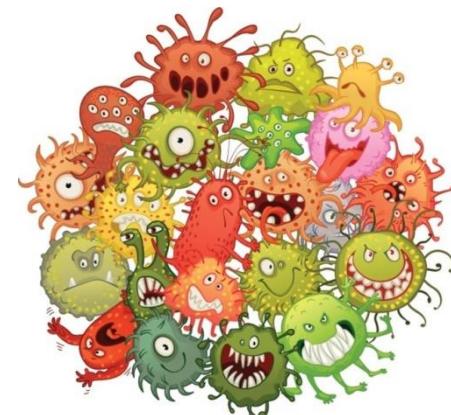


# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = ?$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = ?$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$

$$a_3 = a_2 \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$

$$a_3 = a_2 \cdot 3$$

...

$$a_n = ?$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$

$$a_3 = a_2 \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$

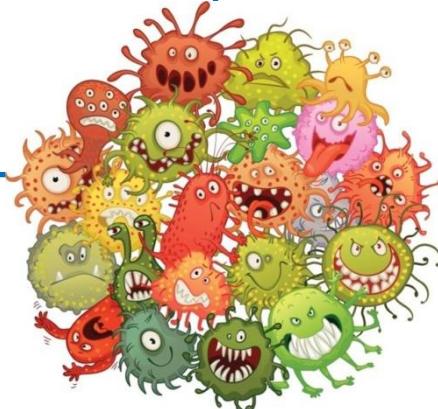
$$a_3 = a_2 \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$

**Relación de recurrencia** para el número de bacterias:

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_1 \cdot 3$$

$$a_3 = a_2 \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = (a_0 \cdot 3) \cdot 3$$

$$a_3 = a_2 \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

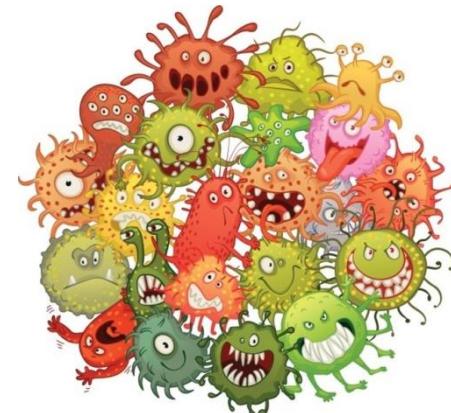
$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = a_2 \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = (a_0 \cdot 3^2) \cdot 3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = a_0 \cdot 3^3$$

...

$$a_n = a_{n-1} \cdot 3$$



# Recurrencias

---

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = a_0 \cdot 3^3$$

...

$$\mathbf{a_n = a_0 \cdot 3^n}$$



# Recurrencias

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas n horas
- **Resolver la recurrencia**
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = a_0 \cdot 3^3$$

...

$$a_n = a_0 \cdot 3^n$$

**La recurrencia:**

$$a_n = a_{n-1} \cdot 3$$

**se soluciona con la fórmula cerrada:**

$$a_n = a_0 \cdot 3^n$$



# Recurrencias

El número de bacterias de una colonia se triplica cada hora. Inicialmente hay 100 bacterias.

- Determinar una relación de recurrencia para el número de bacterias después de transcurridas  $n$  horas
- Resolver la recurrencia
- Indicar la cantidad de bacterias que habrá después de 10 horas

$$a_0 = 100$$

$$a_1 = a_0 \cdot 3$$

$$a_2 = a_0 \cdot 3^2$$

$$a_3 = a_0 \cdot 3^3$$

...

$$a_n = a_0 \cdot 3^n$$

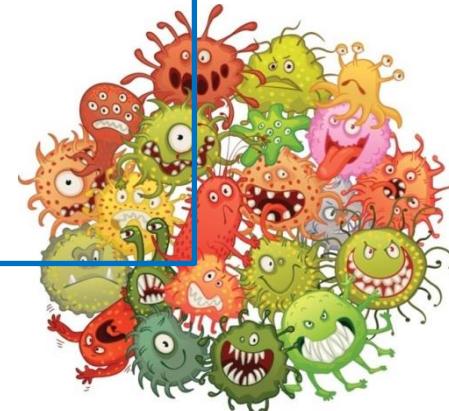
La recurrencia:

$$a_n = a_{n-1} \cdot 3$$

se soluciona con la fórmula cerrada

$$a_n = a_0 \cdot 3^n$$

$$a_{10} = 100 \cdot 3^{10}$$



# Recurrencias

---

Indique la salida de la operación misterio(4)

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 7 | 5 | 6 | 1 | 2 |

# Recurrencias

---

Indique la salida de la operación misterio(4)

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 7 | 5 | 6 | 1 | 2 |

7 5 6 1 2  
7 5 6 1  
7 5 6  
7 5  
7

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

¿Qué recurrencia  
representa el tiempo de  
cómputo de este programa?

# Recurrencias

---

```
public void misterio(int n){    ← T(n)
    if (n==0)
        System.out.print(datos[n]);
    else {
        for(int j=0;j<=n; j=j+1){
            System.out.print(datos[j]);
        }
        System.out.print("\n");
        misterio(n-1);
    }
}
```

# Recurrencias

---

```
public void misterio(int n){    ← T(n)
    if (n==0)
        System.out.print(datos[n]);
    else {
        for(int j=0;j<=n; j=j+1){
            System.out.print(datos[j]);
        }
        System.out.print("\n");
        misterio(n-1);
    }
}
```

¿Dónde se hace el  
**llamado recursivo?** y  
qué subproblema  
resuelve

# Recurrencias

---

```
public void misterio(int n){    ← T(n)
    if (n==0)
        System.out.print(datos[n]);
    else {
        for(int j=0;j<=n; j=j+1){
            System.out.print(datos[j]);
        }
        System.out.print("\n");
        misterio(n-1);    ← T(n-1)
    }
}
```

# Recurrencias

```
public void misterio(int n){    ← T(n)
    if (n==0)
        System.out.print(datos[n]);
    else {
        for(int j=0;j<=n; j=j+1){
            System.out.print(datos[j]);
        }
        System.out.print("\n");
        misterio(n-1);    ← T(n-1)
    }
}
```

Se imprimen n valores

# Recurrencias

---

```
public void misterio(int n){    ←      T(n) = n + T(n-1)
    if (n==0)
        System.out.print(datos[n]);
    else {
        for(int j=0;j<=n; j=j+1){
            System.out.print(datos[j]);
        }
        System.out.print("\n");
        misterio(n-1);
    }
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + n$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + n$$

$$T(0) = ?$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        System.out.print("\n");  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + n$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

Indique la salida de la operación misterio(4)

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 7 | 5 | 6 | 1 | 2 |

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

Indique la salida de la operación misterio(4)

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 7 | 5 | 6 | 1 | 2 |

2 1 6 5 7

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

¿Qué recurrencia  
representa el **tiempo de  
cómputo** de este programa?

# Recurrencias

---

```
public void misterio(int n){    ← T(n)
    if (n==0)
        val = datos[n]
        System.out.print(val);
    else {
        System.out.print(datos[n]);
        misterio(n-1);
    }
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

←  $T(n)$

¿Dónde se hace el  
**llamado recursivo** y qué  
**subproblema** resuelve?

# Recurrencias

---

```
public void misterio(int n){           ← T(n)
    if (n==0)
        val = datos[n]
        System.out.print(val);
    else {
        System.out.print(datos[n]);
        misterio(n-1);           ← T(n-1)
    }
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

←  $T(n)$

} Se imprime 1 valor

←  $T(n-1)$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

←

$$T(n) = ?$$

}

Se imprime 1 valor

←

$$T(n-1)$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```



$$T(n) = 1 + T(n-1)$$

} Se imprime 1 valor



$$T(n-1)$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```



$$T(n) = 1 + T(n-1)$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + 1$$

$$T(0) = ?$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        val = datos[n]  
        System.out.print(val);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
    }  
}
```

$$T(n) = T(n-1) + 1$$

$$T(0) = 2$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
        misterio(n-1);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
        misterio(n-1);  
    }  
}
```

$$T(n) = 2T(n-1) + 1$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
        misterio(n-2);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n-1);  
        misterio(n-2);  
    }  
}
```

$$T(n) = T(n-1) + T(n-2) + 1$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n/3);  
        misterio(n/2);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n/3);  
        misterio(n/2);  
    }  
}
```

$$T(n) = T(n/3) + T(n/2) + 1$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n/5);  
        misterio(n/5);  
        misterio(n/3);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        System.out.print(datos[n]);  
        misterio(n/5);  
        misterio(n/5);  
        misterio(n/3);  
    }  
}
```

$$T(n) = 2T(n/5) + T(n/3) + 1$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        misterio(n/5);  
        misterio(n/5);  
        misterio(n/3);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int n){  
    if (n==0)  
        System.out.print(datos[n]);  
    else {  
        for(int j=0;j<=n; j=j+1){  
            System.out.print(datos[j]);  
        }  
        misterio(n/5);  
        misterio(n/5);  
        misterio(n/3);  
    }  
}
```

$$T(n) = 2T(n/5) + T(n/3) + n$$

$$T(0) = 1$$

# Recurrencias

---

```
public void misterio(int i, int j, int b){  
    int medio=Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
    }  
    else{  
        if (datos[medio]<b)  
            misterio(medio,j,b);  
        if (datos[medio]>b)  
            misterio(i,medio,b);  
    }  
}
```

# Recurrencias

---

```
public void misterio(int i, int j, int b){  
    int medio=Math.round ((i+j)/2);  
    if (datos[medio]==b){  
        System.out.println("Encontrado");  
    }  
    else{  
        if (datos[medio]<b)  
            misterio(medio,j,b);  
        if (datos[medio]>b)  
            misterio(i,medio,b);  
    }  
}
```

$$T(n) = T(n/2) + 1$$

$$T(0) = 1$$