

## Ejercicio 3.16

- El resultado de la monitorización de la actividad de una aplicación informática que está siendo ejecutada dentro de un servidor dedicado a streaming de vídeo se muestra a continuación (nótese que hay información no disponible). Como información adicional, el perfil de llamadas indica que todos los procedimientos son llamados únicamente desde el programa principal `main` (que solo se ejecuta una vez y cuyo tiempo propio de ejecución se puede despreciar), excepto `ordena`, que solo es llamado desde el procedimiento `procesa`.

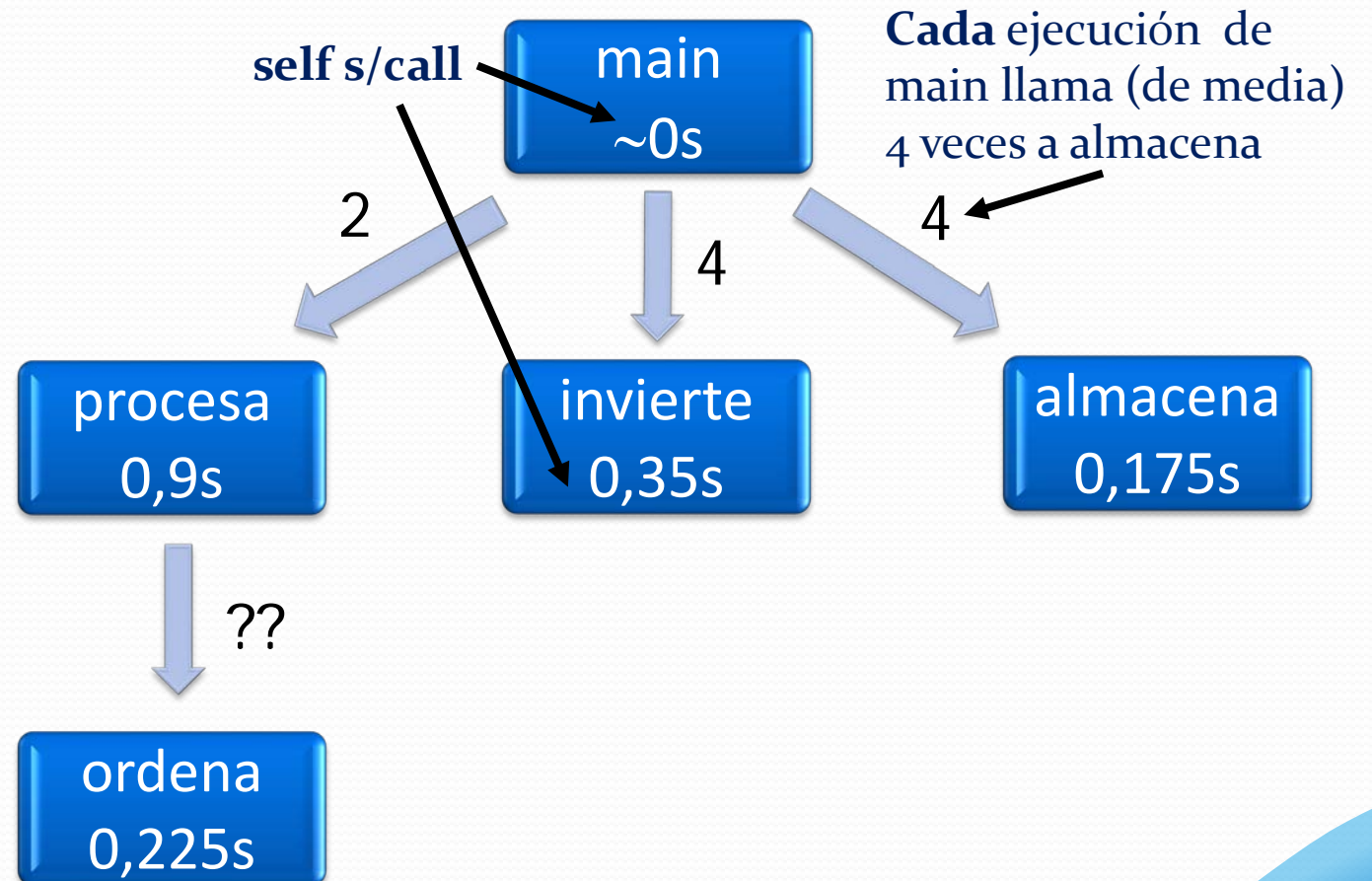
| % time | cumulative seconds | self seconds | calls | self ms/call | total ms/call | name     |
|--------|--------------------|--------------|-------|--------------|---------------|----------|
| XX     | XX                 | 1,8          | XX    | 225          | 225           | ordena   |
| XX     | XX                 | XX           | 2     | 900          | XX            | procesa  |
| XX     | XX                 | 1,4          | 4     | 350          | 350           | invierte |
| XX     | XX                 | XX           | 4     | 175          | XX            | almacena |

- a) Complete la información no disponible en la tabla (marcada como “xx”). ¿Cuánto tiempo de CPU consume la aplicación?

# Paso 1: Intentamos obtener el Grafo de Llamadas

- ... todos los procedimientos son llamados únicamente desde el programa principal `main` (que solo se ejecuta una vez y cuyo tiempo propio de ejecución se puede despreciar), excepto `ordena`, que solo es llamado desde el procedimiento `procesa`.

| calls | self<br>ms/call | total<br>ms/call | name     |
|-------|-----------------|------------------|----------|
| xx    | 225             | 225              | ordena   |
| 2     | 900             | xx               | procesa  |
| 4     | 350             | 350              | invierte |
| 4     | 175             | xx               | almacena |



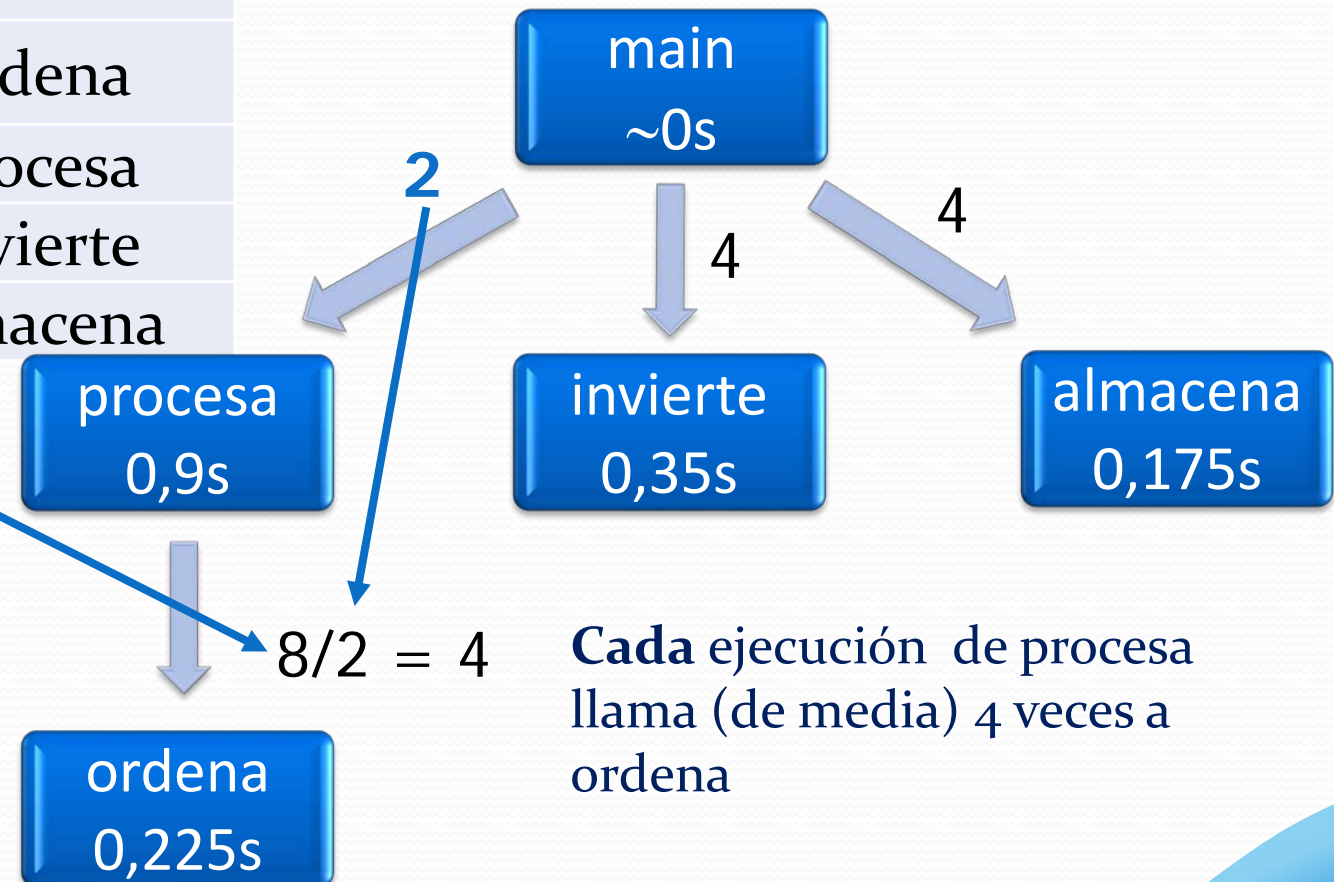
## Paso 2: Empezamos a rellenar la tabla

$$\text{self seconds} = \text{calls} * \frac{\text{self ms/call}}{1000 \text{ ms/s}} \quad \Rightarrow \quad \text{calls} = \frac{\text{self seconds}}{\frac{\text{self ms/call}}{1000 \text{ ms/s}}}$$

| % time | cumulative seconds | self seconds | calls           | self ms/call | total ms/call | name     |
|--------|--------------------|--------------|-----------------|--------------|---------------|----------|
| XX     | XX                 | 1,8          | $1,8/0,225 = 8$ | 225          | 225           | ordena   |
| XX     | XX                 | 1,8          | 2               | 900          | xx            | procesa  |
| XX     | XX                 | 1,4          | 4               | 350          | 350           | invierte |
| XX     | XX                 | 0,7          | 4               | 175          | xx            | almacena |

## Paso 3: Ya podemos completar el Grafo de Llamadas

| nds | calls | self<br>ms/call | total<br>ms/call | name     |
|-----|-------|-----------------|------------------|----------|
|     | 8     | 225             | 225              | ordena   |
|     | 2     | 900             | xx               | procesa  |
|     | 4     | 350             | 350              | invierte |
|     | 4     | 175             | xx               | almacena |



## Paso 4: Continuamos rellorando la tabla: cumulative s

- **Cumulative seconds de una función:** La suma acumulada de los segundos consumidos (CPU) por el código propio de dicha función y por el de las funciones que aparecen encima de ella en la tabla.

| % time | cumulative seconds      | self seconds | calls | self ms/call | total ms/call | name     |
|--------|-------------------------|--------------|-------|--------------|---------------|----------|
| XX     | 1,8                     | 1,8          | 8     | 225          | 225           | ordena   |
| XX     | $1,8+1,8 = 3,6$         | 1,8          | 2     | 900          | xx            | procesa  |
| XX     | $1,8+1,8+1,4 = 5,0$     | 1,4          | 4     | 350          | 350           | invierte |
| XX     | $1,8+1,8+1,4+0,7 = 5,7$ | 0,7          | 4     | 175          | xx            | almacena |

La aplicación consume 5,7s de CPU

## Paso 5: Continuamos rellenando la tabla: %time

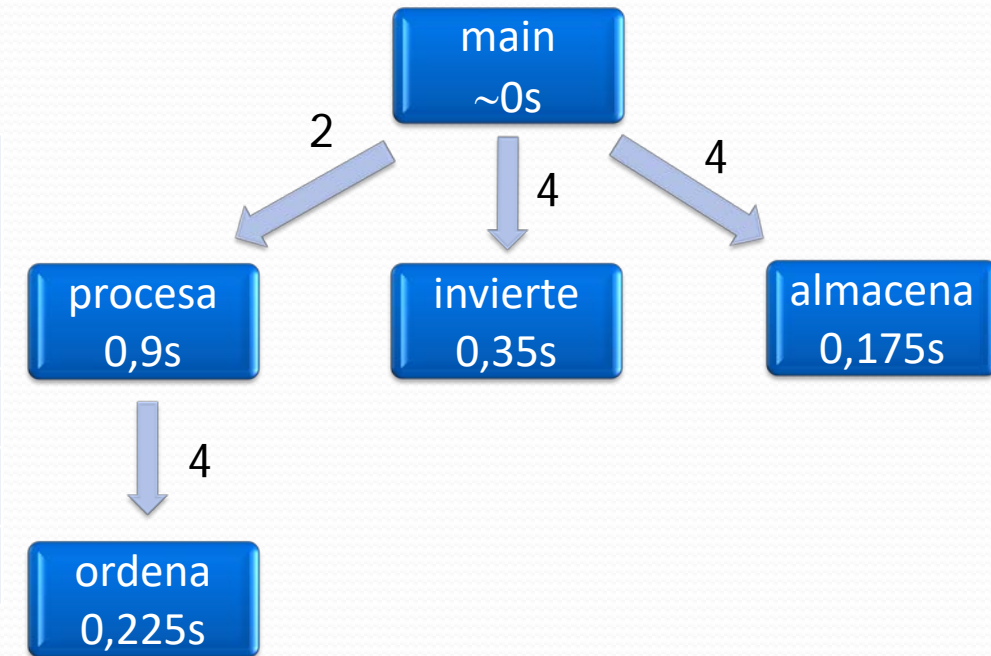
- **% time:** Tanto por ciento del tiempo total de CPU del programa que usa el código propio de la función(código propio es el que pertenece a la función y no a las funciones a las que llama).
  - En nuestro caso: tiempo total de CPU del programa = 5,7s.

| % time                                      | cumulative seconds | self seconds | calls | self ms/call | total ms/call | name     |
|---|--------------------|--------------|-------|--------------|---------------|----------|
| $\left(\frac{1,8}{5,7}\right) * 100 = 31,6$ | 1,8                | 1,8          | 8     | 225          | 225           | ordena   |
| $\left(\frac{1,8}{5,7}\right) * 100 = 31,6$ | 3,6                | 1,8          | 2     | 900          | xx            | procesa  |
| $\left(\frac{1,4}{5,7}\right) * 100 = 24,6$ | 5,0                | 1,4          | 4     | 350          | 350           | invierte |
| $\left(\frac{0,7}{5,7}\right) * 100 = 12,3$ | 5,7                | 0,7          | 4     | 175          | xx            | almacena |

## Paso 6: Acabamos la tabla: total ms/call

- **Total s/call:** tiempo (CPU) medio de ejecución de cada llamada a la función (contando tanto el tiempo del código propio como el de las funciones a las que llama).
  - En nuestro caso, se expresa en ms en lugar de en segundos.

| cumulative seconds | self seconds | calls | self ms/call | total ms/call | name     |
|--------------------|--------------|-------|--------------|---------------|----------|
| 1,8                | 1,8          | 8     | 225          | 225           | ordena   |
| 3,6                | 1,8          | 2     | 900          | 1800          | procesa  |
| 5,0                | 1,4          | 4     | 350          | 350           | invierte |
| 5,7                | 0,7          | 4     | 175          | 175           | almacena |

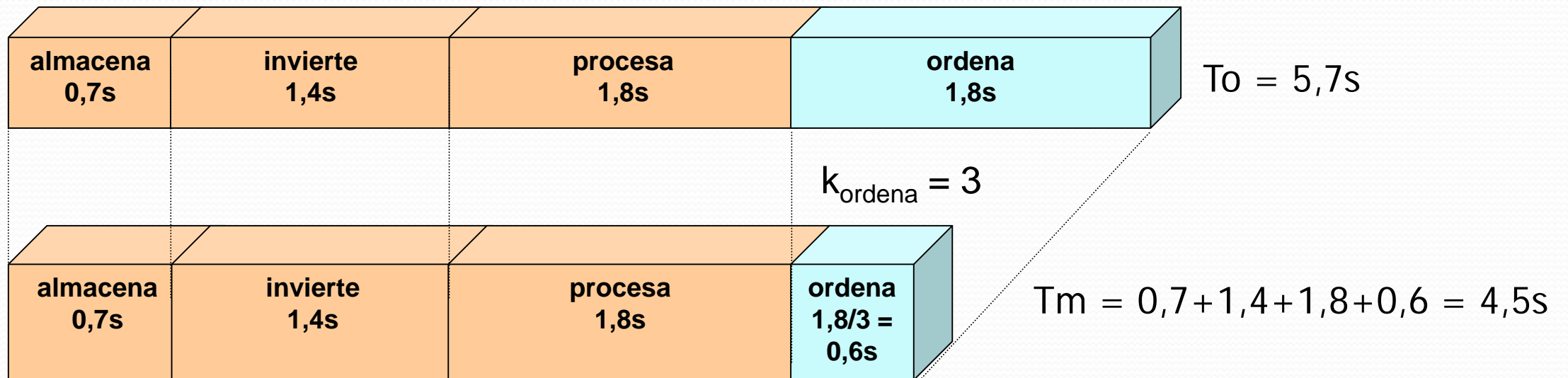


$$\frac{\text{total ms}}{\text{call}}(\text{procesa}) = \frac{\text{self ms}}{\text{call}}(\text{procesa}) + 4 * \frac{\text{total ms}}{\text{call}}(\text{ordena}) = 900\text{ms} + 4 * 225\text{ms} = 1800\text{ms}$$



## Ejercicio 3.16, apartado b)

- b) Determine la ganancia en velocidad (speedup) que se obtendría si reemplazamos el procedimiento ordena por otro 3 veces más rápido. Exprese esa ganancia en velocidad también como tanto por ciento de mejora.



$$S = \frac{v_m}{v_o} = \frac{T_o}{T_m} = \frac{5,7}{4,5} = 1,27$$

El % de mejora obtenido sería:  $(S - 1) * 100 = 27\%$

Otra forma (Ley de Amdahl): 
$$S = \frac{1}{1 - f + \frac{f}{k}} = \frac{1}{1 - 0,316 + \frac{0,316}{3}} = 1,27$$