Realizado por:
- Francisco José Cotán López
- Ignacio Vellido Expósito

1. Usando la notación O, determinar la eficiencia de los siguientes segmentos de código:

```
int n,j; int i=1; int x=0;
 do{
    j=1;
    while (j <= n){
        j=j*2;
        x++;
    }
    i++;
  }while (i<=n);
```

O(1)

O(1)

O(1)

O(1)

O(1)

O(1)

O($\log_2$ n)

O(n) × O($\log_2$ n)

O(n × $\log_2$ n)

```
int n,j; int i=2; int x=0;
 do{
     j=1;
     while (j <= i){
         j=j*2;
         x++;
     }
     i++;
  }while (i<=n);
```

O(1)

O(1)

O(1)

O(1)

O(1)

O(1)

O($\log_2$ i)

O(n + $\log_2$ n!)

O($\log_2$ n!)

2. Para cada función $f(n)$ y cada tiempo t de la tabla siguiente, determinar el mayor tamaño de un problema que puede ser resuelto en un tiempo t (suponiendo que el algoritmo para resolver el problema tarda $f(n)$ microsegundos, es decir, $f(n) \times 10^{-6}$ sg.)

f(n)	***t***				
	1 sg.	1 h.	1 semana	1 año	1000 años
$\log_2 n$	$\approx 10^{300000}$	∞	∞	∞	∞
n	10^6	$\approx 3{,}6 \times 10^9$	$\approx 6 \times 10^{11}$	$\approx 3{,}15 \times 10^{13}$	$\approx 3{,}15 \times 10^{16}$
$n \log_2 n$	$\approx 6 \times 10^4$	$\approx 1{,}33 \times 10^8$	$\approx 10^{10}$	$\approx 7 \times 10^{11}$	$\approx 6 \times 10^{14}$
n^3	100	1532	8456	$\approx 3{,}15 \times 10^4$	$\approx 3{,}15 \times 10^5$
2^n	19	31	39	44	54
$n!$	9	12	14	16	18

Notas

Ejercicio 1:

- $\sum_{i=1}^{n}(\log_2 n) = n \times \log_2 n$

- $\sum_{i=2}^{n}(\log_2 i + 1) = \sum_{i=1}^{n}(\log_2 i + 1) - (\log_2 1 + 1) =$

$$= \sum_{i=1}^{n}(\log_2 i) + \sum_{i=1}^{n}(1) - 1 = \log_2(1+2+\cdots+n) + n - 1 = n + \log_2 n! - 1$$

Ejercicio 2:

1 hora	=	3600	segundos
1 semana	=	604.800	segundos
1 año	=	$3{,}15 \times 10^{7}$	segundos
1000 años	=	$3{,}15 \times 10^{10}$	segundos

Derivando de la fórmula: T(n) = nº de operaciones × velocidad del procesador (s^{-1})
Donde T(n) es el tiempo (en segundos) que tarda en hacer el número de operaciones.

En los casos en los que no se ha podido despejar ha sido a base de prueba y error (sustituciones)

Fórmula para n: $n^{\underline{o}}\ de\ operaciones = \frac{tiempo\ en\ segundos}{velocidad\ del\ procesador}$

Fórmula para n^3: $n^{\underline{o}}\ de\ operaciones = \sqrt[3]{\frac{tiempo\ en\ segundos}{velocidad\ del\ procesador}}$

Fórmula para $\log_2 n$: $n^{\underline{o}}\ de\ operaciones = 2^{\frac{tiempo\ en\ segundos}{velocidad\ del\ procesador}}$

(Los resultados eran tan grandes que se ha aproximado a infinito)