

Práctica 3

Juan Llinares Mauri
jlm109@alu.ua.es

1. Práctica 3 - Complejidad temporal: Cálculo analítico

1.1. Ejercicio 2

```
1 unsigned exercise2 (unsigned m) {  
2     unsigned i=1, p=0;  
3     while (i <= m) {  
4         unsigned q=0, j=i;  
5         while (j > 0) {  
6             j--;  
7             q++;  
8         }  
9         p+=q;  
10        i*=3;  
11    }  
12    return p;  
13 }
```

El tamaño del problema viene dado por m . No hay mejor ni peor caso, pues se trata de un `unsigned` sin contenido interno como puede ser el vector del ejercicio 1.

Realizamos la tabla (1)

Los pasos los podemos interpretar como

$$p_i = 1 + i_k * j = 1 + 3^{k-1} * j = 1 + 3^{\log_3(m)-1} * j = 1 + (m-1) * j = 1 + jm - j$$

, siendo del orden de $O(m)$. Entonces, el programa queda con complejidad:

$$c_i(n) = O(\log m) + O(m) = O(m)$$

Iteración while (i<=m)	i	Pasos
1	1	1
2	3	$1 + 3j$
3	9	$1 + 9j$
4	27	$1 + 27j$
...
$k = \log_3(m)$	$i_k = 3^{k-1}$	$p_i = 1 + i_k * j$

Table 1. Pasos ejercicio 2.