



Instituto Politécnico Nacional Escuela Superior de Cómputo



Análisis de Algoritmos

Práctica 2: Funciones Recursivas vs Iterativas.

Profesor: Dr. Benjamín Luna Benoso.

Grupo: _____

Semestre 2018-1

1. Implementar la sucesión de Fibonacci mediante un algoritmo recursivo y mediante un algoritmo iterativo.

- i) Mostrar los algoritmos con la notación vista en clase (pseudocódigo).
- ii) Demostrar formalmente que el algoritmo iterativo tiene orden lineal.
- iii) Mostrar mediante gráficas que la proposición ii) es cierta.
- iv) A partir de gráficas experimentalmente, proponer el orden de complejidad para el algoritmo recursivo.

2. Implementar el siguiente algoritmo:

Algoritmo $S(n)$

Input: Un entero positivo n .

Output: La suma de los primeros n cubos

```
1  if  $n = 1$  return 1
2  else return  $S(n - 1) + n * n * n$ 
```

i) A partir de gráficas, proponga una función $g(n)$ tal que $T(n) \in O(g(n))$ con $T(n)$ el tiempo computacional del algoritmo.

ii) Implemente al algoritmo de manera iterativa, y a partir de gráficas, proponga una función $g(n)$ tal que $T(n) \in O(g(n))$ con $T(n)$ el tiempo computacional del algoritmo en el caso iterativo.

iii) Calcule el tiempo computacional del algoritmo recursivo e iterativo de manera formal. Compare sus resultados.

Resolver los siguientes problemas:

1. Calcular el orden de complejidad de algoritmo de la burbuja (**BubbleSort**).

BubbleSort(A)

```
1  for  $i = 1$  to  $i \leq A.length - 1$ 
2      for  $j = A.length$  downto  $j \geq i + 1$ 
3          if  $A[j] < A[j - 1]$ 
4              exchange  $A[j]$  with  $A[j - 1]$ 
```