PARCIAL TEORICO

PRESENTADO POR:

JUAN CAMILO BAZURTO ARIAS

PRESENTADO A:

SEBASTIAN CAMILO MARTINEZ REYES

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO

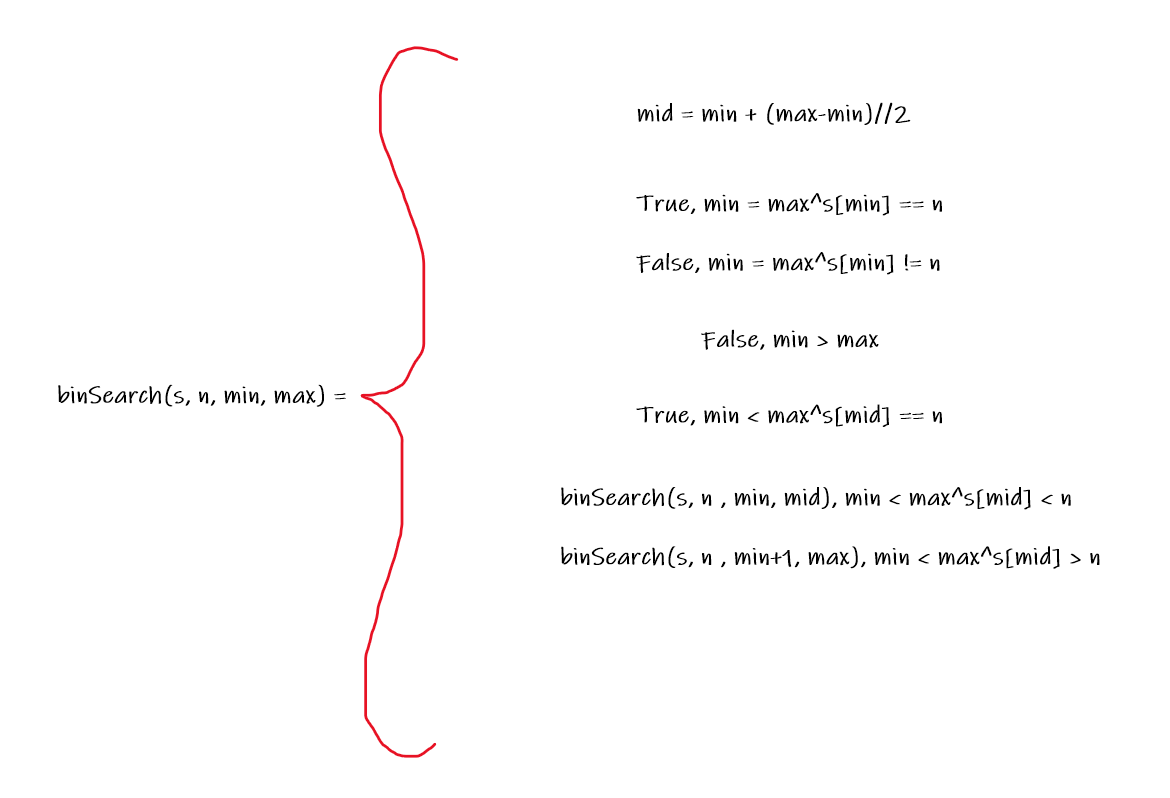
ALGORITMOS Y ESTRUCTURAS DE DATOS

PROGRAMA DE INGENIERÍA DE SISTEMAS

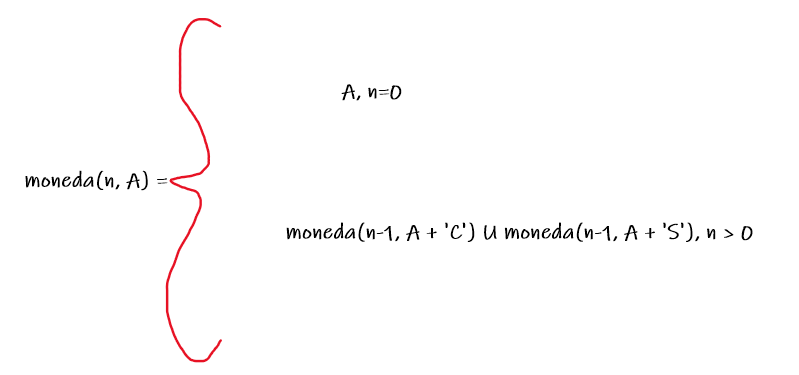
BOGOTÁ D.C.

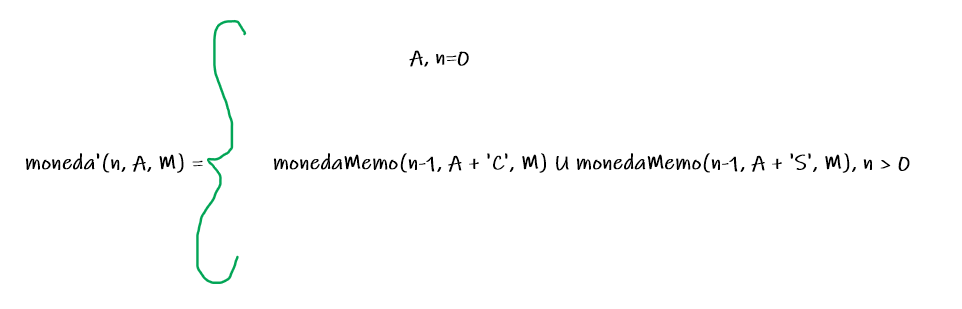
2021 – 1

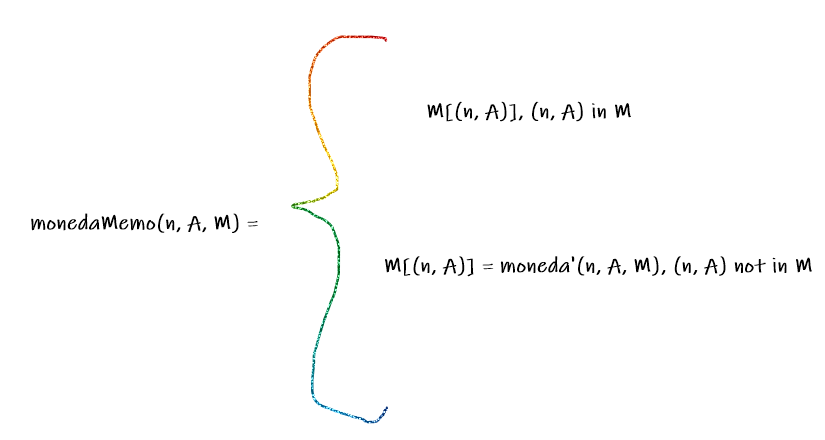
1. Dada una lista ordenada de elementos A = x0, x1, x2 . . . xi . . . xn. Escriba una función recurrente utilizando la estrategia de dividir y conquistar, donde dado un elemento q. La función responda si el elemento q se encuentra en la lista A.



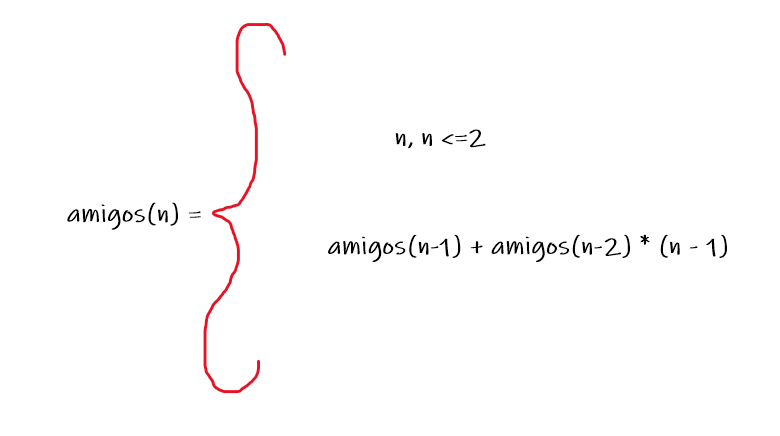
1. Escriba una función f(n . . .) recurrente que satisfaga lo siguiente: Suponga que Juan lanza una moneda al aire N veces, cuyos valores posibles son ’C’(Cara)y ’S’(S el l o ) para cada evento independiente de lanzamiento. Pedro dese a llevar registro de todos los eventos de lanzamiento que produce Juan. Por lo que para el caso de N= 3, El registro de pedro es el conjunto A={CCC, CCS, CSC, CSS, SCC, SCS,SSC, SSS}, escriba una función que genere el conjunto A para cualquier N. Aplique el método de memorización por Top-Down^







1. Suponga que hay n amigos, quienes pueden permanecer solos o ser emparejados con otro amigo. Cada amigo puede ser emparejado una sola vez. Encuentre el número de formas en que los amigos pueden quedar solos o emparejados.



1. Ordene de 1 a N las siguientes funciones características de crecimiento asintótico.

|  |  |
| --- | --- |
| f(x) | Orden |
|  | 5 |
| nlg(n) | 3 |
|  | 4 |
| ln(n) | 1 |
| n | 2 |

1. Marque la casilla, solo si la afirmación que la acompaña es verdadera.

 Ω(n) = {f(n) : existen constantes positivas c y n0tales que 0 <= f(n) <=cg(n) para todo n >= n0

Falso, porque la función omega representa el mejor de los casos, por lo tanto, la función que tiene la constante debería estar f(n) <= 0

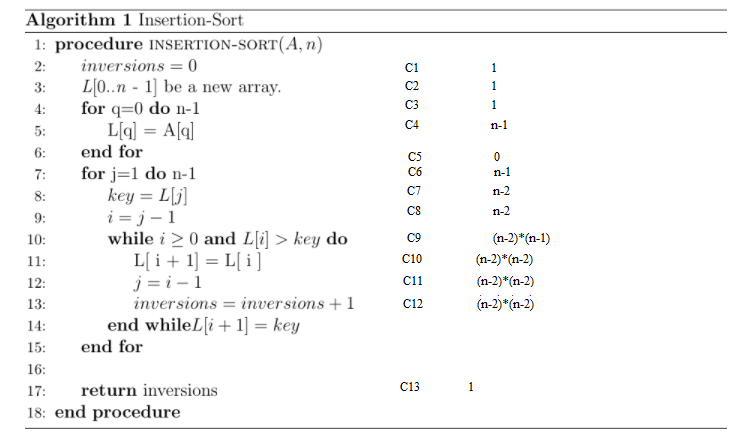
 El teorema maestro aplica para la solución de recurrencias de la forma T (n) =aT (n/b) + f(n)

Verdadero, por definición de teorema maestro.

Un algoritmo se considera correcto si para un subconjunto de la entrada. Entrega una salida correcta.

Falso, porque el algoritmo debe retornar todo el problema y no solo una solución parcial.

1. Dado el algoritmo de insertion-sort realice una estimación del tiempo de ejecución en donde demuestre que T (n) es de la forma T (n) = an2+ bn + c
2. procedure insertion-sort(A, n)
3. 2: inversions = 0
4. 3: L[0..n - 1] be a new array.
5. 4: for q=0 do n-1
6. 5: L[q] = A[q]
7. 6: end for
8. 7: for j=1 do n-1
9. 8: key = L[j]
10. 9: i = j − 1
11. 10: while i ≥ 0 and L[i] > key do
12. 11: L[ i + 1] = L[ i ]
13. 12: j = i − 1
14. 13: inversions = inversions + 1
15. 14: end whileL[i + 1] = key
16. 15: end for
17. 16:
18. 17: return inversions
19. 18: end procedure
20. procedure insertion-sort(A, n)
21. 2: inversions = 0
22. 3: L[0..n - 1] be a new array.
23. 4: for q=0 do n-1
24. 5: L[q] = A[q]
25. 6: end for
26. 7: for j=1 do n-1
27. 8: key = L[j]
28. 9: i = j − 1
29. 10: while i ≥ 0 and L[i] > key do
30. 11: L[ i + 1] = L[ i ]
31. 12: j = i − 1
32. 13: inversions = inversions + 1
33. 14: end whileL[i + 1] = key
34. 15: end for
35. 16:
36. 17: return inversions
37. 18: end procedure



T(n) = (a)n^2 + bn + c