ESTRUCTURA DE DATOS 2
Código ST0247

Laboratory practice No. 5
Divide to Conquer and Dynamic Programming

Jhonatan Sebastián Acevedo Castrillón
Universidad Eafit
Medellín, Colombia
jsacevedoc@eafit.edu.co

Manuel Alejandro Gutiérrez Mejía
Universidad Eafit
Medellín, Colombia
magutierrm@eafit.edu.co

3) Practice for final project defense presentation

3.1 We used matrix as the main data structure and at the same time we help us with auxiliary arrangements to follow a path order. The algorithm works based on the theory of dynamic programming and divide to conquer, remembering in a certain way the previous solutions if in the following iterations were necessary, iterations that finish when the cost of the path guided by the minimum weights exceeds a maximum cost previously extracted in the same way and the initial node was not reached with all the nodes visited before.

3.2 The algorithm has to do $n^2\ 2^n$ operations, where n is the number of nodes in the graph, therefore with 50 nodes it would do about $2.8x10^{18}$ operations.

3.3 We started to represent the graph as a adjacency matrix, in that matrix we save the initial position in which all the radioactive waste is found. After that we choose an implementation of the Held Karp algorithm because we need the shortest path between Karolina and the radioactive waste.

3.4 O(2^n*n^2).

3.5 n is the length of the adjacency matrix.

4) Practice for midterms

4.1 Calle Casa

PhD. Mauricio Toro Bermúdez
Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473

	∈	c	a	l	l	e
∈	0	1	2	3	4	5
c	1	0	1	2	3	4
a	2	1	0	1	3	3
s	3	2	1	1	2	3
a	4	3	2	2	1	3

Mamá-Madre

	∈	m	a	d	r	e
∈	0	1	2	3	4	5
m	1	0	1	2	3	4
a	2	1	0	1	2	3
m	3	2	1	1	2	3
á	4	3	2	2	2	3

4.2 **1)**O(lenx*leny) 2) table[lenx][leny]
4.3 **1) a)**O(n) 2) a) c1:n+c2
4.4 **1)c)** O(2^n) y se optimiza con programación dinámica.
4.5 **1) c)** T(n/2)+C que es O(log n) **2)** a[mitad] 3) a[mitad]+1;
4.6 **1)** scm[i]=1 **2)** scm[i] = scm[j]+1; **3)** max = scm[i];
4.7 **1)** d[i][j]; **2)** d[k][j]; **3)** d[i][k;**] 4)** O(n^2)

PhD. Mauricio Toro Bermúdez
Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473