

Carla Pérez Gavilán Del Castillo, A01023033

17 de abril de 2020

Prof. Vicente Cubells

Análisis y Diseño de Algoritmos

PROBLEMA 7:

El mejor camino posible con Dijkstra Algoritmo

Una compañía de ferrocarriles sirve n estaciones S_1, \dots, S_n y trata de mejorar su servicio al cliente mediante terminales de información. Dadas una estación origen S_o y una estación destino S_d , un terminal debe ofrecer (inmediatamente) la información sobre el horario de los trenes que hacen la conexión entre S_o y S_d y que minimizan el tiempo de trayecto total.

TÉCNICA: Programación Dinámica

Cálculo de Complejidad

NOTA: Se calculará la complejidad de la función que calcula el mejor camino posible, las funciones de impresión y de transformación de el csv al problema se consideran despreciables

FUNCIÓN:

`template <class T>`

`void TrainTime<T>::bestpath(T graph[V][V], T begin, T end)`

No.	Instrucción	Operaciones Elementales	Ciclo	Veces	Total Simplificado
1	<code>int times[V];</code>	0	0	0	0

	bool visited[V]; int parent[V];				
2	for (int i = 0; i < V; i++)	3	1	V+1	2V+2
3	times[i] = INT_MAX;	2	V	1	2V
4	visited[i] = false;	2	V	1	2V
5	parent[begin] = -1;	2	1	1	2
6	times[begin] = 0;	2	1	1	2
7	for (int i = 0; i < V - 1; i++)	4	1	V-1	$1+V-1+V-2 = 2V-2$
8	int a = min(times, visited);	10V-7	V-2	1	$10V^2 - 27V + 14$
9	visited[a] = true;	2	V-2	1	V-2
10	for (int j = 0; j < V; j++) {	3	V-2	V-1	$V-2(1+V+V-1)$ $V-2(2V)$ $2V^2 - 4V$
11	if (!visited[j] && graph[a][j] && times[a] +	12	V-2 (V-1)	1	$12V^2 - 36V + 24$

	graph[a][j] < times[j])				
12	parent[j] = a;	2	V-2 (V-1)	1	$2V^2 - 6V + 4$
13	times[j] = times[i] + graph[a][j];	6	V-2 (V-1)	1	$6V^2 - 18V + 12$
14	if(j==end){	1	V-2 (V-1)	1	$V^2 - 3V + 2$
COMPLEJIDAD TOTAL		Peor de los casos			$T(V) = 33V^2 - 85V + 58$

**Nota: consideramos la función de print es despreciable

FUNCIÓN AUXILIAR:

template <class T>

int TrainTime<T>::min(int times[], bool visited[])

{

	Instrucción	Operaciones Elementales	Veces	Ciclos	Total Simplificado
1	iint min = INT_MAX; int index = 0;	2	1	1	2

Carla Pérez Gavilán Del Castillo, A01023033

17 de abril de 2020

Prof. Vicente Cubells

Análisis y Diseño de Algoritmos

2	for (int i = 0; i < V; i++)	3	V	1	$1+V+V-1$ $2V$
3	if (!visited[i] && times[i] <= min)	5	1	V-1	$5V-5$
4	min = times[i]; index = i;	3	1	V-1	$3V-3$
5	return index;	1	1	1	1
Complejidad Total en el Peor caso					$10V-7$

CÁLCULO DE COTA ASINTÓTICA

$$\lim_{V \rightarrow \infty} \frac{33V^2 - 85V + 58}{V^2} = 363$$

Como el resultado es una constante sabemos que pertenecen al mismo orden de complejidad:

$$O(V) = V^2$$

$$O(n) = n^2$$