

Laboratorio Nro. 3: Backtracking

Mateo Ramírez Hernández

Universidad Eafit
Medellín, Colombia
marami26@eafit.edu.co

Juan Camilo Echeverri S.

Universidad Eafit
Medellín, Colombia
jechev60@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

1. Los algoritmos más importantes son

El algoritmo de Dijkstra resuelve el problema de ruta más corta de fuente única.

El algoritmo Bellman-Ford resuelve el problema de fuente única si los pesos de borde pueden ser negativos.

Un algoritmo de búsqueda * resuelve la ruta más corta de un solo par utilizando heurísticas para intentar acelerar la búsqueda.

El algoritmo Floyd-Warshall resuelve todos los caminos más cortos de los pares.

El algoritmo de Johnson resuelve todos los caminos más cortos de los pares, y puede ser más rápido que Floyd-Warshall en gráficos dispersos .

El algoritmo Viterbi resuelve el problema más corto de la ruta estocástica con un peso probabilístico adicional en cada nodo.

2.

VALOR DE N	FUERZA BRUTA Seg(aprox)	BACKTRACKING Seg(aprox)
4	0.01	0.002
8	0.5	0.06
16	Mas de 5 min	Mas de 5 min
32	Mas de 5 min	Mas de 5 min
N	$O(n!)$	$O(n^n)$

3. Para recorrer grafos DFS es mas conveniente cuando el grafo tiene una profundidad corta , pero una gran anchura, mientras que el BFS es más conveniente al revés es decir cuando el grafo tiene una profundidad larga, pero poca anchura

DOCENTE MAURICIO TORO BERMÚDEZ

Teléfono: (+57) (4) 261 95 00 Ext. 9473. Oficina: 19 - 627

Correo: mtorobe@eafit.edu.co

4. **Greedy BFS:** expande el primer sucesor del padre. Después de que se genera un sucesor

A*: a partir de un nodo específico del grafo, construye un árbol de rutas a partir de ese nodo, ampliando las rutas un paso a la vez, hasta que una de sus rutas termina en el nodo del objetivo predeterminado.

Uniform Cost search: es una variante del algoritmo de Dijkstra, y este consiste en recorrer el vértice no visitado con la distancia más baja, calcula la distancia a través de él a cada vecino no visitado y actualiza la distancia del vecino si es más pequeña.

5. $O(n+m)$

6.

7. *N se refiere a los vértices del grafo y M se refiere a los bordes del grafo*

4) Simulacro de Parcial

1.

- a. $N-a, a, b, c$
- b. $\text{Res, solucionar}(n-b, a, b, c) + 1$
- c. $\text{Res, solucionar}(n-c, a, b, c) + 1$

2.

- a. Graph.length
- b. $V, \text{graph}, \text{path}, \text{pos}$
- c. $\text{Graph}, \text{path}, v$

3.

DFS:

- $0 \rightarrow 3\ 7\ 4\ 2\ 1\ 5\ 6$
- $1 \rightarrow 0\ 3\ 7\ 4\ 2\ 6\ 5$
- $2 \rightarrow 1\ 0\ 3\ 7\ 5\ 4\ 6$
- $3 \rightarrow 7$
- $4 \rightarrow 2\ 1\ 0\ 3\ 7\ 5\ 6$
- $5 \rightarrow$
- $6 \rightarrow 2\ 1\ 0\ 3\ 7\ 5\ 4$
- $7 \rightarrow$

BFS:

- $0 \rightarrow 3\ 4\ 2\ 7\ 1\ 6\ 5$
- $1 \rightarrow 0\ 5\ 2\ 3\ 4\ 6\ 7$
- $2 \rightarrow 1\ 4\ 6\ 0\ 5\ 3\ 7$

- 3 -> 7
- 4 -> 2 1 6 0 5 3 7
- 5 ->
- 6 -> 2 1 4 0 5 3 7
- 7 ->

5. Lectura recomendada (opcional)

- a) Título
- b) Ideas principales
- c) Mapa de Conceptos

6. Trabajo en Equipo y Progreso Gradual (Opcional)

- a) Actas de reunión
- b) El reporte de cambios en el código