

SDA - Backtracking + Alți Algoritmi

Daniel Chiș - 2021, UPB, ACS, An I, Seria AC

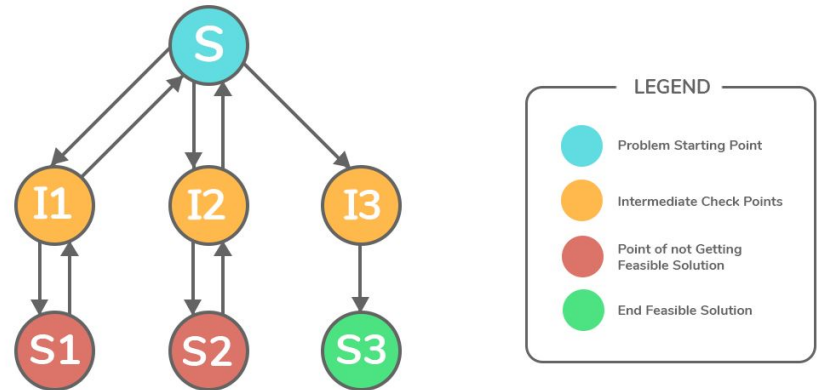


Backtracking

Concept

Backtracking se referă la folosirea unui algoritm de tip recursiv pentru a rezolva o problemă ce admite soluții parțiale. Se începe cu o soluție locală și se avansează până la construirea unei soluții globale. Dacă o rută pe care am început să construim nu duce nicăieri, atunci ne întoarcem -backtrack- și se avansează pe altă cale.

Backtracking



Algoritm

- Se alege un punct de start
- Cât timp problema nu este rezolvată:
 - Pentru fiecare cale (parte din soluție disponibilă) de la punctul de start:
 - Se verifică dacă ruta aleasă este bună (nu invalidează condiția de rezolvare)
 - Dacă da, se adaugă la soluția în construcție și se apelează recursiv algoritmul pentru restul problemei
 - Dacă apelul recursiv reușește să găsească o soluție completă, algoritmul se încheie. Dacă nu, ruta aleasă se elimină din soluție și se alege alta.
- Dacă niciuna din căile disponibile nu este bună, nu există soluție (pentru punctul de start ales).

Sudoku

- Matrice 9x9
- Pe coloana, linie, sub-matrice de 3x3 trebuie să existe un număr o singură dată

Funcție de unassigned - verifică dacă o celulă nu are valoare

Funcție is_safe - verifică dacă poate să adauge o valoare de la 1-9

Funcție solve - apelează primele 2 funcții prin backtracking pentru a rezolva sudoku

								8
1	8				2	3		
	6			5	7			1
	7		9	6				
	9		7		4		1	
				8	1		4	
6			2	4			8	
		4	5				9	3
5								



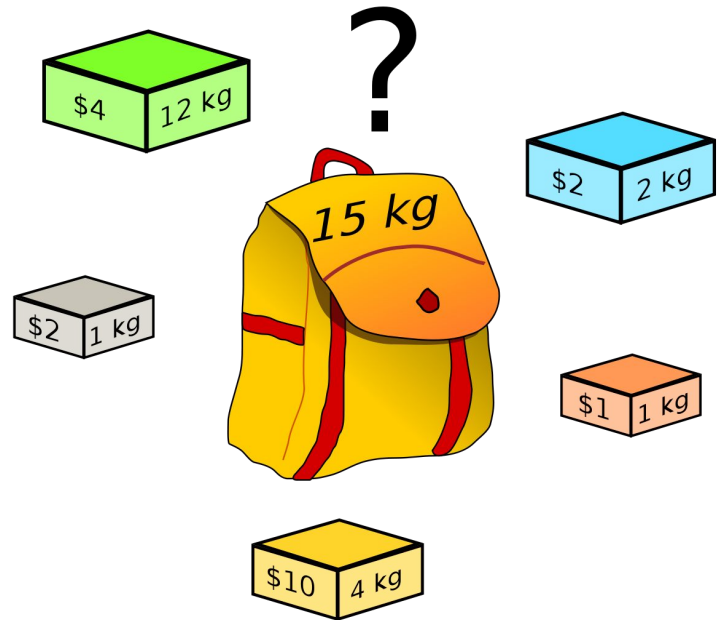
Altı Algoritmi

Problema rucsacului

Este o problemă de optimizare care poate fi rezolvată atât recursiv cât și prin programare dinamică.

Problemă:

- Capacitate maximă X
- Pot fi adăugate obiecte de greutate a, b, c cu diferite valori
- Se cere valoare maximă care poate fi atinsă



Algoritm A*

Algoritmul A* este unul foarte bun pentru a găsi drumul critic, traversări de grafuri. Acesta calculează cel mai bun drum bazat pe o euristică (algoritm de aproximare). Un exemplu de euristică este distanța euclidiană:

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Noțiuni esențiale

- Nod - posibile poziții de deplasare
- Tranziție - deplasarea de la un nod la altul
- Nod destinație
- Cost - distanța dintre un nod și altul
- $g(n)$ - costul exact al drumului dintre nodul de start și nodul următor
- $h(n)$ - costul estimat folosind o euristică de la nodul următor la destinație
- $f(n)$ - costul minim de la un nod la vecin
- $f(n) = g(n) + h(n)$



Predictii

Regresie Liniară - Linear Regression

Link: <https://www.analyticsvidhya.com/blog/2020/04/machine-learning-using-c-linear-logistic-regression/>



Exerciții

Exerciții

1. Rezolvați Sudoku-ul de la slide-ul 6 folosind backtracking. 3p
2. Rezolvați problema rucsacului de la slide-ul 8. 3p
3. Găsiți cel mai bun drum folosind A* search pentru următorul traseu. 3p

