

Mohó algoritmus

Mi az algoritmus célja?:

A mohó algoritmusokat akkor alkalmazzuk, amikor egymásra épülő döntéseket kell hozni, hogy elérjük az optimális megoldást. Az alapelv az, hogy minden lépésben helyileg legjobbnak tűnő döntést hozunk. Például a pénzváltás során, ha egy adott összeget kell kifizetni a lehető legkevesebb érme felhasználásával, mindig a legnagyobb címletű érméből használunk, amíg a fennmaradó összeg ezt megengedi. Számos probléma esetén a mohó módszer gyorsabban nyújt optimális megoldást, mint például a dinamikus programozás. Azonban nem mindig könnyű meghatározni, hogy a mohó megközelítés hatékony lesz-e adott problémára.

Az algoritmus lépései:

Rendezzük az eseményeket a befejezési idejük szerint növekvő sorrendbe. Kezdjük egy üres listával, amelyben a kiválasztott eseményeket fogjuk tárolni. Haladjunk végig az eseményeken, és válasszuk ki azokat, amelyek kezdési ideje később van, mint az utolsóként kiválasztott esemény befejezési ideje.

Az algoritmus magyarázata:

- `events.sort(key=lambda x: x[1])`: Az eseményeket a befejezési idő szerint rendezi.
 - `last_end_time`: Nyomon követi az utolsó kiválasztott esemény befejezési idejét.
 - Az `if start >= last_end_time` feltétellel ellenőrzi, hogy egy esemény ütközik-e az előző kiválasztott eseménnyel.
 - Ha nem ütközik, hozzáadja a kiválasztott események listájához.
-