Domača naloga #5: dinamično programiranje 1
Priprave na računalniške olimpijade 2018/19

Tomaž Hočevar
tomaz.hocevar@fri.uni-lj.si

A. k-Tree

Že v besedilu naloge so dobro opisani podproblemi, ki jih rešujemo. Ker ima drevo neskončno rekurzivno strukturo, je vseeno, v katerem vozlišču se nahajamo. Edina pomembna podatka sta vsota n, ki jo moramo dobiti, in zastavica $D = \{\text{da}, \text{ne}\}$, ki nam pove, ali moramo uporabiti neko povezavo z vrednostjo d, ali ne.

Naj $f(n, D)$ predstavlja iskano število načinov. Za rešitev podproblema moramo upoštevati vseh k povezav. Ločimo dva primera glede na to, ali je povezava enaka d. Imamo $O(n)$ podproblemov, za rešitev posameznega pa potrebujemo $O(k)$ časa za skupno časovno zahtevnost $O(nk)$.

$$f(n, D) = f(n - d, \text{ne}) + \sum_{x \neq d} f(n - x, D)$$

B. Shaass and Bookshelf

Naloga je rešljiva tudi brez dinamičnega programiranja, če vam "dobra vila" prišepne, da je treba vertikalno postaviti n_1 knjig debeline 1 in n_2 knjig debeline 2. Na vrh želimo zložiti čim krajše knjige, zato spodaj izberemo n_1 najdaljših debeline 1 in n_2 najdaljših debeline 2. Če preostanek lahko zložimo na vrh, imamo potencialno rešitev. Ker dobre vile nimamo, preverimo vseh $O(n^2)$ kombinacij števil n_1, n_2 za skupno časovno zahtevnost $O(n^3)$.

Opazimo lahko, da rešitev v prejšnjem odstavku obravnava možnosti kot so npr. $(n_1 = 10, n_2 = 5), (n_1 = 8, n_2 = 4), (n_1 = 6, n_2 = 3), \ldots$, ki imajo vse isto vsoto debelin, zanima pa nas samo najboljša med njimi. Zastavimo si lahko podproblem $f(i, d)$ na sledeč način. Ukvarjamo se samo s prvimi i knjigami, pri čemer nas zanima, kakšna je maksimalna vsota širin knjig, katerih debeline se seštejejo v d (preostanek bo namreč treba zložiti na vrh). Naloga je zelo podobna problemu nahrbtnika. Rešitev lahko izračunamo z obravnavo podproblemov $f(n, x)$ za vse možne širine spodnjega nivoja x. Ta algoritem ima časovno zahtevnost $O(n^2)$.

$$f(i, d) = \max(f(i - 1, d), f(i - 1, d - t_i) + w_i)$$

C. Playing Piano

Zastavimo so podproblem $f(i, p) = $ da/ne ... ali je možno odigrati melodijo od i-te note do konca, če i-to odigramo s prstom p. V skladu z razmerjem med notama a_i in a_{i+1} moramo obravnavati največ 5 primerov naslednjega prsta q, ki ga bomo uporabili za igranje note $i + 1$.

$$f(i, p) = \begin{cases} \bigvee_{q>p} f(i+1, q), & \text{če } a_{i+1} > a_i \\ \bigvee_{q<p} f(i+1, q), & \text{če } a_{i+1} < a_i \\ \bigvee_{q \neq p} f(i+1, q), & \text{če } a_{i+1} = a_i \end{cases}$$

Poleg rešitve podproblema $f(i, p)$ si shranimo še naslednji prst q, ki nas je pripeljal do te rešitve, da lahko kasneje rekonstruiramo neko iskano zaporedje prstov.