Univerzitetni programerski maraton 2025
1. kolo — rešitve nalog

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

21. februar 2025

Zig Zag

Vzorec je periodičen, zato si lahko pripravimo osnovni vzorec v velikosti 4x8 in nato v vrstici i in stolpcu j izpišemo znak $v[i \,\%\, 4, j \,\%\, 8]$.

Min nasledniki

Dolžini števil lahko poravnamo z dodatkom vodilnih ničel. Skupna predpona števil a in b bo ves čas enaka. Na prvem mestu (npr. i), kjer se razlikujeta, pa izberemo manjšo od števk a_i in b_i. V preostanku se bo na vsakem mestu zagotovo pojavila kakšna ničla, zato lahko rezultat dopolnimo z ničlami. Po potrebi moramo poskrbeti še za brisanje vodilnih ničel v rezultatu.

Cesarjevo novo vozilo

Problema se lahko lotimo z bisekcijo. Če obstaja rešitev za vozilo velikosti x, bo tudi za manjše. Tako smo prevedli problem na iskanje rešitve za neko fiksno velikost vozila a. Prav bi nam prišla informacija, ali se vozilo dane velikosti lahko nahaja z levim zgornjim robom na koordinati (r, c). To lahko učinkovito počnemo tako, da si vnaprej izračunamo kumulativne vsote, iz katerih lahko nato s štirimi operacijami izračunamo vsoto poljubnega pravokotnika na polju. Ko vemo, katere pozicije so dovoljene, samo še preverimo dosegljivost levega zgornjega z desnim spodnjim vogalom, npr. z iskanjem v širino.

Texas hold'em poker

To je povsem implementacijska naloga, kjer nas ne skrbi učinkovitost, ampak samo pravilnosti naše rešitve. Za vsakega igralca lahko petim skupnim kartam dodamo še njegovi dve in preverimo vseh 21 kombinacij, kako izbrati 5 izmed 7 kart. Sedaj moramo še ovrednotiti kvaliteto izbranih petih kart, tako da natančno sledimo navodilom. Prav nam pridejo pomožne funkcije za ugotavljanje kompleta enakih barv, lestvice in za razbijanje seznama v skupine enakih kart glede na vrednosti. Slednja lahko vrne seznam parov (število kart, vrednost karte) urejen od večjih proti manjšim skupinam in nato od večjih vrednosti proti manjšim. Npr. seznam [(3,A), (2,9)] predstavlja full house asov s parom devetk. Posebej pazljivi moramo biti na primer, kjer as nastopa kot najnižja karta v lestvici.

Kontrolne vsote

Naloga je sestavljena iz dveh delov. Najprej moramo ugotoviti verjetnost posamezne kontrolne števke, nato pa izračunati, kakšno je pričakovano število potez, da zberemo vse kontrolne števke. Prvi problem rešujemo z dinamičnim programiranjem, kjer nam $p(i, m)$ predstavlja verjetnost, da je po obravnavi prvih i števk ostanek pri deljenju z 11 enak m. Naslednja števka bo $x \in [1..9]$, zato lahko $p(i, m)/10$ prištejemo k verjetnosti $p(i + 1, (m(a_{i+1}x^2 + b_{i+1}x + c_{i+1}))\%11)$. Verjetno kontrolne vsote m je $q(m) = p(n, m\%10)$.

Za izračun pričakovanega števila potez bomo prav tako uporabili dinamično programiranje. Tokrat bodo stanja opisana s podmnožico kontrolnih števk, ki nam še manjkajo. Vrednost $f(S)$ naj predstavlja pričakovano število potez za pridobitev manjkajočih števk iz množice S. Probleme lahko rešujemo od manjših proti večjim množicam S. Na vsaki potezi imamo možnost dobiti neko staro ali pa novo števko: $f(S) = 1 + \sum_{m \in S} q(m) f(S) + \sum_{m \notin S} q(m) f(S - \{m\})$. Če enačbo malo obrnemo, lahko $f(S)$ izrazimo v odvisnosti od $f(S')$, kjer je $S' \subset S$, kar imamo že izračunano.

Žrebec

Posebej lahko obravnavamo primere, ko je $a = 0$ ali $b = 0$. Očitno sta relevantni samo razliki $d_x = x_2 - x_1$, $d_y = y_2 - y_1$. Ugotovimo lahko, da nima smisla, da izvajamo tako premik (a, b) kot tudi $(-a, -b)$ — smiseln je samo en od njiju, faktor pa bo morda negativen. Tako imamo štiri različne smeri, določiti pa moramo število premikov v vsaki od teh smeri. Za vsako dimenzijo (x in y) dobimo enačbo s štirimi neznankami, ko združimo sistem teh dveh enačb pa nam ostane linearna diofantska enačba (LDE) s tremi spremenljivkami oblike $Ax + By + Cz = D$, kjer so A, B, C, D konstante, ki jih dobimo iz vrednosti a, b, d_x, d_y. Rešimo jo tako, da jo prevedemo na dve LDE z dvema spremenljivkama: $Gw + Cz = D, (A/G)x + (B/G)y = w'$, kjer je $G = \gcd(A, B)$, w' pa ena rešitev prve enačbe. LDE z dvema spremenljivkama $Sx + Ty = U$ rešimo s pomočjo razširjenega Evklidovega algoritma. Ta nam za konstanti S, T vrne x in y, da velja $Sx + Ty = \gcd(S, T) = g$. Rešitev LDE dobimo tako, da skaliramo rešitev razširjenega Evklidovega algoritma za U/g. Pri tem moramo paziti, da dobljena rešitev ni prevelika. Zmanjšamo jo lahko tako, da rešitev x', y' spremenimo v $x' - kT, y' + kS$ za poljuben oz. primeren k. Četrto manjkajočo spremenljivko dobimo iz ene od začetnih enačb s štirimi spremenljivkami.