

Elektrijada 2022

Algoritmi i strukture podataka - Rešenja

Zadatak 1

Lako je pogrešiti u ovakvom zadatku i deliti loptice na dva jednaka dela, međutim tako sa jednog koraka možemo preći sa P loptica na $P/2$.

Ako ipak stavimo $P/3$ na levi i $P/3$ na desni tas, u jednom merenju možemo odrediti u kojoj trećini je teža loptica, jer ako vaga ostane u ravnoteži to bi značilo da je teža loptica u onoj trećini koja nije bila na vagi. Time smo jednim merenjem prešli sa P na $P/3$, što je i optimalan način za merenje.

Odatle izvodimo da je rešenje najmanje K za koje je $\frac{P}{3^k} \leq 1$, odnosno gde je $P \leq 3^k$.

Pošto je za primer pod c) potreban veliki stepen trojke, da ne bismo računali puno množenja, korisno je iskoristiti tehniku binarnog stepenovanja i izračunati:

$$3^2 = 9, \quad 3^4 = (3^2)^2 = 81, \quad 3^8 = (3^4)^2 = 6561, \\ 3^{16} = (3^8)^2 = 43046721, \quad 3^{32} = (3^{16})^2 = 1853020188851841$$

Za primer pod b) vidimo da 3^8 možemo podeliti sa 3 i dobiti opet broj veći od 2023, pa je tu rešenje 7, dok za primer pod c) vidimo samo prebrojavanjem cifara da moramo pomnožiti još jednom 3^{32} sa 3 i (čak i bez računanja rezultata tog množenja) zaključujemo da je rešenje 33.

Zadatak 2

Možemo pokazati da sigurno ne možemo rasporediti više od $2/3 * N$ dama (gde je N dimenzija table).

Najpre, ako sa X_d osnačimo red u kojoj se neka dama nalazi (indeksirano od 0), sa Y_d kolonu, a sa Z_d udaljenost od dijagonale imamo da je $Z_d = N - (X_d + Y_d)$ i imamo da je $X_d + Y_d + Z_d = N$.

X_d mora biti različito za sve dame, kao i Y_d i Z_d , jer ukoliko je isto to bi značilo da se međusobno napadaju. To znači da kada rasporedimo K dama, suma X koordinata mora biti barem

$$0 + 1 + 2 + \dots + K - 1 = \frac{(K-1)*K}{2}. \text{ Isto važi i za } Y \text{ i } Z \text{ koordinate.}$$

Kako je $X_d + Y_d + Z_d = N$, takođe znači da suma svih koordinata svih dama je tačno $K \cdot N$, pa odatle imamo da mora da važi

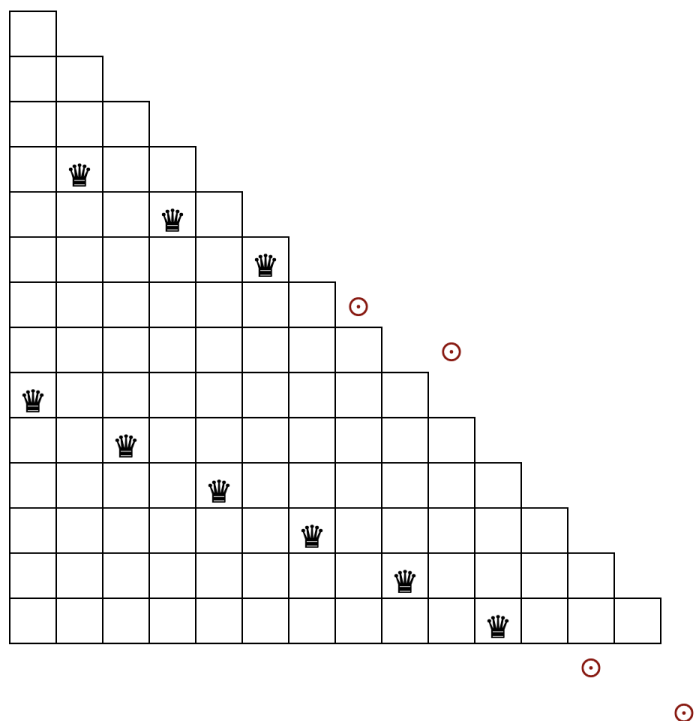
$$K \cdot N = \sum_{i=1}^n (X_i + Y_i + Z_i) \geq \min_sum(X) + \min_sum(Y) + \min_sum(Z) = 3 \cdot \frac{(K-1) \cdot K}{2}$$

Odatle sledi,

$$K \cdot N \geq 3 \cdot \frac{(K-1) \cdot K}{2} \Rightarrow N \geq 3 \cdot \frac{(K-1)}{2} \Rightarrow K \leq \frac{2 \cdot N}{3} + 1$$

Dobili smo da nikako ne možemo imati više od $\frac{2}{3} \cdot N + 1$, a to nas upućuje da pokušamo da nađemo raspored koji sadrži tačno toliko dama, jer bi to sigurno onda bio i maksimalan broj.

Jedan od takvih rasporeda je kao na slici:



(crvene tačkice označavaju način na koji se raspored nastavlja za veće dimenzije table)

Zadatak 3

Zadatak se radi dinamičkim programiranjem.

Ako sa $D[i][j]$ označimo broj načina tako da odaberemo za farabanje na pravilan način cigle iz prvih i redova, dok u i -tom redu odaberemo ciglu j .

Onda je $D[i][j] = \text{suma}(D[i-1][p])$ za svako p koje je $p \leq j$ i $p \leq j+1$. Bazni slučaj je $D[1][j] = 1$. Treba obratiti pažnju da svaki parni red ima jednu ciglu manje. Rešenje je $\sum_{j=1}^n D[n][j]$.

U primeru pod a) treba videti da se brojevi uglavnom ponavljaju, $D[2][j] = 2020$ za svako j , pa je onda $D[3][1] = 2020 \cdot 2020$, $D[3][2..M-1] = 2020 \cdot 2019$, $D[3][M] = 2020 \cdot 2020$.

Rešenje je onda $(2020 + (2019 \cdot 2020) + 2020) \cdot 2020 = 2021 \cdot 2020 \cdot 2020$.

U primeru pod b) možemo primetiti da kada je $M=3$, nebitno koliko je N imamo samo 2 validna načina za bojenje.

Za primere pod c) i d) da bismo ubrzali računanje, treba primetiti ne moramo izračunati celu matricu $D[i][j]$. Naime, $D[i][j] = D[i][M-i]$, pa svaki red je dovoljno računati do pola. Takođe, kada smo izračunali do reda $N/2$, možemo zaključiti da je broj načina za računanje u gornjoj polovini simetričan sa brojem načina u donjoj polovini, pa nas to dovodi do rešenja da je

$$\text{Rešenje} = \sum_{j=1}^n D[n/2][j]^2$$

Zadatak 4

Ovo je zadatak gde se u odnosu na dati primer treba opredeliti za pristup rešavanju, pa ćemo zato sva 3 primera rešavati na drugačiji način.

- a) Možemo primetiti da je slika simetrična po vertikali, to znači da je dovoljno izbrojati trouglove na jednoj polovini pa pomnožiti sa dva, i onda dodati trouglove koji seku vertikalu kakvih vidimo da nema puno.

Metodičan način za brojanje ovih trouglova je da obeležimo sva temena, zapišemo sve trojke brojeva (sve potencijalne trouglove) i proverimo da li postoje linije između tih temena. Tako smo sigurniji da nećemo propustiti nijedan trougao.

- b) Za početak je potrebno da izračunamo koliko trouglova postoji u skroz popunjenoj "piramidi" (trouglu skroz popunjenom drugim trouglovima kao na slici) dimenzija $N \times N$ (gde je N broj horizontalnih nivoa piramide). Ovo možemo raditi tako što odvojeno trouglove kojima je vrh "na gore" i onima kojima je vrh "na dole".

Od onih koji su na gore njih ima:

dimenzije 1: $1+2+3+\dots+N = T_n$

dimenzije 2: $1+2+3+\dots+N-1 = T_{n-1}$

dimenzije 3: $1+2+3+\dots+N-2 = T_{n-2}$

...

dimenzije N : $1 = T_1$

Brojevi označeni sa T_x su poznati kao trougaoni brojevi, a može se pokazati da je

$$T_n = \frac{N(N+1)}{2}, \text{ a da je } \sum_{i=1}^n T_i = \frac{N(N+1)(N+2)}{6}$$

Kod trouglova okrenutih na dole za svaku dimenziju se broj "nivoa" smanjuje za 2, pa bi njihov broj bio:

Dimenzija 1: $1+2+3+\dots+N-1 = T_{n-1}$

Dimenzije 2: $1+2+3+\dots+N-3 = T_{n-3}$

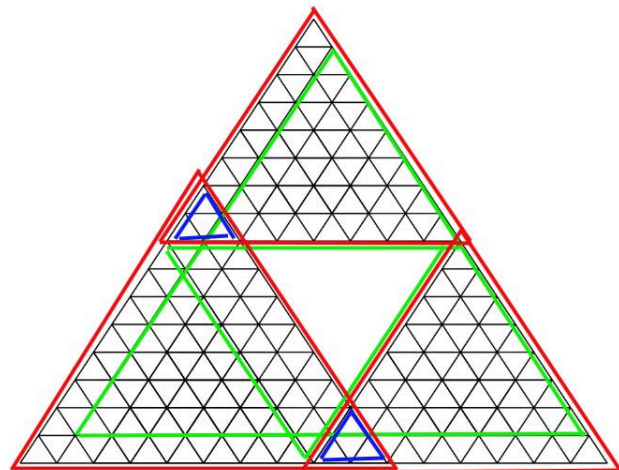
Dimenzije 3: $1+2+3+\dots+N-5 = T_{n-5}$

... dimenzije $N/2 = 1 = T_1$

Dimenzije $N/2 + 1, \dots, N: 0$

Kako nam je potrebno najviše do 10 dimenzije, ove sume možemo odvojeno izračunati i sabrati za trouglove okrenute na gore i na dole.

Sada je potrebno uvideti koje sve popunjene trouglove imamo (crveni), i oduzeti od njih one koje smo računali po 2 puta (plavi). Zatim akda to izračunamo, potrebno je dodati sve one koji u potpunosti sadrže prazni beli trougao u sredini (neki od njih su označeni zelenom bojom).



- c) Potrebno je uočiti da se slika sastoji od nekoliko pravih unutar pravougaonika, od kojih se skoro svaka seče sa svakom drugom (osim jedne). Ako izuzmemu tu jednu, od preostalih 6 pravih svake 3 koje čine tačno jedan trougao, i time prave $\binom{6}{3} = \frac{6*5*4}{3*2} = 20$ trouglova. Preostala prava se seče sa 3 prave (2 dodiruje i jednu seče) i sa njima pravi još 3 dodatna trougla.
- Potrebno je ne zaboraviti da i stranice pravougaonika obrazuju neke trouglove, pa tako levu stranicu seku 2 prave koje sa tom stranicom čine još 1 trougao, gornju stranicu seku 5 pravih koje čine još 10 trouglova, desnu stranicu 2 prave koje čine još 1 trougao, i donju stranicu 3 prave koje čine 3 trougla. Uz to imamo i trouglove koje obrazuju po 2 susedne stranice pravougaonika i neke prave, tako imamo da levu+gornju stranicu seče jedna prava, gornju+desnu takođe jedna, i donju+desnu jedna, što sve čini još 3 trougla. Kada sve pažljivo saberemo dobijamo 41 trougao.

Zadatak 5

Iako je zadatak poznat kao "quick-select" algoritam, na papiru nije uvek optimalno raditi kao što bi radio računar.

Za prva dva primera možemo uzeti neki broj iz niza, a zatim izbrojati koliko brojeva u nizu postoji koji su manji od njega. Ukoliko je broj manjih tačno K-1, to znači da nam je baš to traženi broj. Ukoliko je broj manjih manji od K-1, uradićemo istu proveru ali sa nekim manjim brojem, dok ukoliko nam je broj veći od K-1 uradićemo sledeću proveru sa većim brojem.

U drugom možemo brže izbrojati koliko ima manjih/većih jer je niz skoro sortiran i u svakom redu ima isto brojeva.

Kod primera c) i d) niz je veliki i nasumično raspoređen, pa bi prolaženje više puta kroz niz oduzelo puno vremena. Takođe pošto je isti niz za oba primera, najbolje je pokušati u isto vreme tražiti rešenje za oba primera.

Jedan od načina je da posmatramo samo prve dve cifre svakog broja (pošto su svi brojevi u rangu 1000-9999), i da za svake dve odredimo koliko se puta pojavljuje u nizu broj koji počinje sa te dve cifre.

Pošto je u primeru pod c) $K = 22$, očekujemo da traženi broj počinje sa 1 ili 2, a u primeru pod d) broj je središnji u nizu, pa možemo očekivati da će mu prva cifra biti 4,5, ili 6. Ovo znači da samo za prve cifre 1,2,4,5 i 6 nam je bitna i druga cifra, a za ostale nam je bitna samo prva. Ovo će nam ubrzati prolazak i označavanje kroz niz, a omogućiti da saznamo prve dve cifre traženih brojeva. Kada to znamo, potrebno je još jednom proći kroz niz i popisati brojeve koji počinju sa te dve cifre, i naći tačan po redosledu koji nam treba.