

### RADIO

Jedan način da dođemo od frekvencije A do B je da ne koristimo memorirane frekvencije, a za to nam treba broj pritisaka jednak apsolutnoj vrijednosti njihove razlike.

Primijetimo da nema smisla koristiti više od jedne tipke s memoriranim frekvencijama, jer možemo odmah skočiti na onu najbližu frekvenciji B.

Učitavamo jednu po jednu memoriranu frekvenciju te provjeravamo postižemo li bolje rješenje korištenjem te frekvencije. Pri korištenju neke frekvencije potreban broj pritisaka jednak je jedan plus apsolutna vrijednost razlike frekvencije B i memorirane frekvencije.

### CRTALO

Obrađujemo naredbu po naredbu. U svakom trenutku pamtim par brojeva  $(r, s)$  – red i stupac u kojem se nalazi crtalo, a u dvodimenzionalna polja  $H$  i  $V$  zapisujemo je li crtalo kroz određeni kvadratić prošlo horizontalno, odnosno vertikalno.

Za svaku naredbu radimo sljedeće:

- U ovisnosti o naredbi izračunamo par brojeva  $(p, q)$  – red i stupac u kojem će se crtalo nalaziti nakon pomaka
- Ako je kvadratić  $(p, q)$  izvan papira, onda ignoriramo naredbu (prelazimo na sljedeću naredbu)
- Ako je naredba pomaka 'L' ili 'R', onda postavljamo  $H(r,s) = 1$  i  $H(p,q) = 1$
- Ako je naredba pomaka 'U' ili 'D', onda postavljamo  $V(r,s) = 1$  i  $V(p,q) = 1$
- Pomičemo crtalo, tj.  $r$  postaje  $p$ , a  $s$  postaje  $q$

Nakon što obradimo sve naredbe preostaje nam samo za svaki kvadratić  $(i, j)$  ispisati odgovarajući znak i to:

- Znak '.' – ako je  $H(i, j) = 0$  i  $V(i, j) = 0$
- Znak '-' – ako je  $H(i, j) = 1$  i  $V(i, j) = 0$
- Znak '|' – ako je  $H(i, j) = 0$  i  $V(i, j) = 1$
- Znak '+' – ako je  $H(i, j) = 1$  i  $V(i, j) = 1$

Službena rješenja u C-u i Pascalu pokazuju ovaj pristup, a rješenje u jeziku C++ pamti oba podatka za neki kvadratić u jednom cijelom broju (u različitim bitovima).

### BARD

Za svakog mještana potrebno je pratiti koje sve pjesme zna (koristimo dvodimenzionalno polje Booleovih varijabli).

Ako je bard prisutan, pjeva novu pjesmu i svi prisutni mještani nauče tu novu pjesmu. Ako bard nije prisutan, pjevaju se sve pjesme koje zna bilo tko od prisutnih te svi prisutni nauče te pjesme. Pjesme koje se pjevaju možemo izračunati u posebnom polju pomoću jednostavnih logičkih operacija.

Na kraju je potrebno pronaći i ispisati sve mještane koji znaju sve pjesme.

### BUBA

Primijetimo da je međusoban poredak stalagmita (kao i stalaktita) nebitan pa ih možemo sortirati uzlazno po visini.

Sada možemo primjetiti da će buba nakon što poruši prvi stalagmit porušiti i sve nakon njega upravo zbog toga što je svaki stalagmit veći ili jednak prethodnome. Isto vrijedi i za stalaktite.

Dakle, kako bismo izračunali koliko prepreka će buba porušiti na nekoj razini dovoljno je pronaći prvi stalagmit i prvi stalaktit koji će buba porušiti u sortiranim nizovima.

Kako su nizovi sortirani, te brojeve možemo pronaći binarnim pretraživanjem. Vremenska složenost rješenja je  $O(N + H \log N)$ , a memorijska  $O(N)$ . Službeno rješenje u jeziku C++ pokazuje ovaj pristup.

Alternativno, možemo u dva polja sa  $H$  elemenata izračunati koliko ima stalagmita i koliko ima stalaktita većih od neke visine. Vremenska složenost ovakvog rješenja je  $O(N+H)$ , ali je memorijska  $O(H)$ . Primjer implementacije je u C i Pascal kôdovima.