

ProFIIT 2019

zadania problémov pre korešpondenčné kolo súťaže

Fakulta informatiky a informačných technológií
Slovenská technická univerzita v Bratislave

Úvod

Najprestížnejšia banka na svete FIITa banka¹ sa rozhodla pre inovatívny krok: na svojich pobočkách využije robotov, ktoré vedia komunikovať so zákazníkmi.

Po predchádzajúcich skúsenostiach sa na nás banka opäť obrátila s prosbou o pomocou s technickými výzvami, ktoré pri nasadzovaní robotov na pobočky vznikli.

Prajeme vám príjemné zážitky pri vylepšovaní robotov a pohľade do zákulisia FIITa banky. Veľa šťastia pri riešení úloh.

Upozornenie: Pre všetky problémy platí takéto pravidlo: posledný riadok výstupu má byť odriadkovaný presne tak, ako všetky ostatné riadky výstupu (t.j. na konci každého riadku je ENTER).

¹minimálne od prvej písomnej zmienky spred siedmych rokov

Zvuková analýza

Roboty slúžia aj na získanie spätnej väzby od zákazníkov, t.j. ako boli spokojní s návštevou pobočky banky. Zákazník vysloví svoje hodnotenie jednotlivých ukazovateľov ako ochota pracovníkov, rýchlosť vybavenia, apod. Celkové hodnotenie je potom súčet ohodnotení jednotlivých ukazovateľov.

Robot obsahuje podprogram, ktorý predspracuje zvukový záznam s odpoveďou zákazníka do textovej podoby. Podprogram dokáže vytvoriť prepis, kde nie sú medzery ani diakritika, ale zase vie rozpoznať čísla. Prepis potom môže vyzeráť napr. „Ochota10Pomoc6Rychlost8“.

Zo získaných prepisov zvukového záznamu teraz potrebujeme získať súčet všetkých čísel, ktoré sa v zázname nachádzajú. Číslo je súvislá postupnosť po sebe idúcich číslic. Ostatné znaky v prepise slúžia pre naše účely len na oddelenie jednotlivých čísel.

Vstup

Prvý riadok vstupu obsahuje počet prepisov zvukového záznamu P ($1 \leq P \leq 500$). Každý prepis je na samostatnom riadku, má dĺžku 1 až 100 znakov a pozostáva zo znakov malej a veľkej anglickej abecedy a číslic. Čísla nachádzajúce sa v prepisoch sú od 0 po 5 000 vrátane bez zbytočných núl na začiatku.

Výstup

Pre každý zadaný prepis zvukového záznamu vypíšte na samostatný riadok súčet čísel v prepise alebo hlášku „Ziadne cislo nebolo najdene“.

Vzorový vstup

```
5
Ochota10Pomoc6Rychlost8
a12b14c
a1214c
abc
0
```

Vzorový výstup

```
24
26
1214
Ziadne cislo nebolo najdene
0
```

Banko-vklado-mat

Hlavnou výhodou kamennej pobočky banky je možnosť vložiť alebo vybrať si hotovosť. Pre urýchlenie týchto operácií sa banka rozhodla využiť služby robota. Zákazník robotovi povie, koľko peňazí chce vložiť, alebo koľko peňazí potrebuje vybrať a robot ho nasmeruje buď na vkladomat, alebo bankomat.

Robot má vždy správnu informáciu o aktuálnej hotovosti na pobočke. V prípade, keď si chce zákazník vybrať toľko peňazí, že by hotovosť na pobočke klesla pod stanovený minimálny limit, robot upozorní obsluhu, aby takúto kritickú situáciu so zákazníkom vyriešila.

Robotov algoritmus na rozpoznávanie čísel v hovorenom slove funguje spoľahlivo. To, čo niekedy zlyháva, je rozpoznanie, či chce zákazník peniaze vložiť alebo vybrať. Ak sa robot pomýli, zákazník si poklepe na čelo a sám nájde vkladomat alebo bankomat a robot dostane správnu informáciu o stave hotovosti na pobočke. Problém je to však vtedy, ak by robot nesprávne vyhodnotil, že hotovosť na pobočke klesne pod stanovený minimálny limit, t.j. vyvolal by falošnú kritickú situáciu.

Preto je zaujímavá informácia, najmenej koľko zákazníkov musí prísť, aby hotovosť na pobočke dosiahla aspoň takú hodnotu, kedy omyl robota určite nevyvolá falošný poplach (nakoľko existuje limit na výber/vloženie hotovosti pre jedného zákazníka). Samozrejme za predpokladu, že ani jeden zákazník nevyvolá falošný poplach.

Napríklad, ak

- stanovený minimálny limit hotovosti na pobočke je 5 000
- aktuálna hotovosť na pobočke je 9 000
- hodnota hotovosti, kedy omyl robota určite nevyvolá poplach, je 15 000

potom musia prísť aspoň dvaja zákazníci. Napríklad prvý vloží 4 000 (4 001 by už mohlo spôsobiť falošný poplach, ak by robot nesprávne vyhodnotil požiadavku ako výber) a druhý vloží napríklad 2 300. Celkovo tak bude na pobočke 15 300.

Vstup

Na prvom riadku je číslo T ($1 \leq T \leq 500$) vyjadrujúce počet testovacích scenárov.

Pre každý scenár je na samostatnom riadku trojica celých čísel M, H, R ($0 < M < H < R < 1\,000\,000\,000$) oddelených medzerou. M vyjadruje stanovený minimálny limit hotovosti na pobočke, H je aktuálny stav hotovosti na pobočke a R predstavuje výšku hotovosti na pobočke, kedy omyl robota určite nevyvolá falošný poplach.

Výstup

Na výstupe je pre každý testovací scenár na samostatnom riadku najmenší možný počet zákazníkov, po návšteve ktorých stav hotovosti na pobočke bude aspoň R , pričom žiadny zákazník určite nevyvolá poplach.

Vzorový vstup

```
3
5 9 13
10 11 30
100 500 600
```

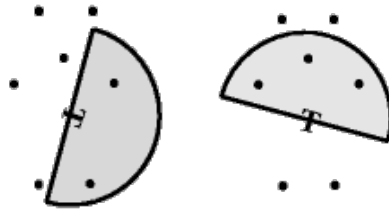
Vzorový výstup

```
1
5
1
```

Natočenie

Pri komunikácii robota s ľuďmi je dôležité natočiť robota tak, aby mal v zornom poli čo najväčší počet ľudí. Potrebujeme nájsť spôsob ako určiť, v ktorom zornom poli s daným dosahom bude robot vidieť najviac ľudí, ak sa môže otáčať.

Napríklad na obrázku vľavo sú v robotovom zornom poli len dvaja ľudia. Po otočení sú to už traja ľudia, čo je maximum pre daný dosah.



Vstup

Vstup má na prvom riadku číslo P ($1 \leq P \leq 1\,100$) reprezentujúce počet úloh.

Každá z úloh má na prvom riadku tri čísla X_r, Y_r, R oddelené medzerou. Čísla X_r, Y_r ($0 \leq X_r, Y_r \leq 1\,000$) reprezentujú polohu robota a reálne číslo R ($0 < R < 300$) s presnosťou na 6 desatinných miest reprezentuje dosah zorného poľa. Na ďalšom riadku je číslo N ($1 \leq N \leq 150$). Nasleduje N riadkov so súradnicami X_c, Y_c ($0 \leq X_c, Y_c \leq 1\,000$) oddelených medzerou, vyjadrujúce polohu ľudí. Všetky súradnice sú navzájom rôzne.

Výstup

Pre každú úlohu vypíšte na samostatný riadok maximálny počet ľudí, ktorí sa môžu nachádzať v (alebo aj na) polkruhu s daným dosahom pri jeho správnom natočení.

Vzorový vstup

```
3
25 25 3.5
7
25 28
23 27
27 27
24 23
26 23
24 29
26 29
```

350 200 2.0
5
350 202
350 199
350 198
348 200
352 200
995 995 10.0
4
1000 1000
999 998
990 992
1000 999

Vzorový výstup

3
4
4

OCR

Robot dokáže rozoznať zamestnancov banky pomocou ich vysačky s menom a číslom zamestnanca. Nie vždy to však funguje spoľahlivo. Preto je potrebné vylepšiť rozpoznanie pridaním špeciálneho atramentu, ktorý je detekovateľný laserom, ale ľudské oko nezbadá rozdiel.

Takýto atrament je veľmi drahý, preto ho chceme použiť čo najmenej. Napríklad pre nasledovný font s piatimi symbolmi:

```
.o. .o. oo. oo. o.o
o.o .o. ..o ..o o.o
o.o .o. .o. oo. ooo
o.o .o. o.. ..o ..o
.o. .o. ooo oo. ..o
```

stačí atrament naniesť len na jeden alebo dva „pixely“ každého symbolu:

```
.#. .o. #o. oo. o.#
#.o .#. ..o ..o o.o
o.o .o. .o. #o. ooo
o.o .o. #.. ..o ..o
.o. .o. ooo #o. ..o
```

Takto je totiž každý symbol jednoznačne identifikovateľný v rámci fontu bez znalosti ako sú zafarbené iné symboly. Napríklad pre symbol 1 stačí jeden „pixel“, pretože na druhom riadku a druhom stĺpci má o len on. Podobne pre symbol 3 sú potrebné dva „pixely“, lebo len kombinácia dvoch o je preňho jedinečná – vyznačená je jedna z možností.

Pre zvolený font je potrebné zistiť čo najmenší počet „pixelov“, ktoré treba zafarbiť špeciálnym atramentom tak, aby bol každý symbol jednoznačne identifikovateľný. Pre každý symbol je dovolené zafarbiť najviac dva „pixely“.

Vstup

Vstup má na prvom riadku číslo P ($1 \leq P \leq 200$) vyjadrujúce počet úloh.

Každá z úloh má na prvom riadku tri čísla N, R, S oddelené medzerou. N ($1 \leq N$) reprezentuje počet symbolov vo fonte, R ($1 \leq R \leq 10$) predstavuje počet riadkov a S ($1 \leq S \leq 80$) počet stĺpcov, z ktorých je zložený jeden symbol. Nasleduje R riadkov, na ktorých je vykreslený font, kde N symbolov je od seba oddelených medzerou. Každý riadok obsahuje najviac 81 znakov.

Výstup

Pre každú úlohu vykreslite font s označenými „pixelmi“ pomocou znaku #. Ak sa nedá zaručiť jednoznačná identifikácia symbolu z fontu na vstupe použitím max. dvoch „pixelov“ na symbol, tak vypíšte „nemozne“.

Ak existuje viacero riešení, zvolte „pixel“ najviac hore a ak je stále viacero riešení, tak „pixel“ najviac vľavo. Podobne postupujte aj ak je potrebné zafarbiť dva „pixely“.

Vzorový vstup

```
3
5 5 3
.o. .o. oo. oo. o.o
o.o .o. ..o ..o o.o
o.o .o. .o. oo. ooo
o.o .o. o.. ..o ..o
.o. .o. ooo oo. ..o
3 2 2
oo oo .o
o. .o o.
1 2 5
.o...
...oo
```

Vzorový výstup

```
.#. .o. #o. oo. o.#
#.o .#. ..o ..o o.o
o.o .o. .o. #o. ooo
o.o .o. #.. ..o ..o
.o. .o. ooo #o. ..o
nemozne
.#...
...oo
```


Šípky

Niektorých robotov využijú na pobočke, kde budú prezentovať bankové produkty, pričom sa budú pohybovať po šípkami vyznačenej trase. Tieto sú na zemi nakreslené ľudským okom neviditeľným atramentom (a navyše sú malé, keďže atrament je drahý).

Priestor označený šípkami môže vyzeráť napríklad takto (znaky v, ^, > a < reprezentujú políčka, na ktorých sa nachádza šípka – dole, hore, vpravo a vľavo):

```
..v
^>.
..<
```

Robota ráno položia na niektorú zo šípiek, a potom sa už celý deň pohybuje v smere posledne spozorovanej šíčky. Šíčku zaregistruje, len keď na nej stojí. Cieľom je, aby sa robot pohyboval len vo vyznačenom priestore.

A práve to je problém. Zistilo sa, že niektoré priestory majú šípky nakreslené tak, že ak by na ne ráno položili robota, ten by z priestoru vybehol. Preto potrebujeme zistiť, koľko najmenej šípok treba opraviť (zmeniť na iné), aby sa to nemohlo stať. Premiestňovať, pridávať, ani uberať šípky nemožno.

Vstup

Na prvom riadku sa nachádza číslo P ($1 \leq P \leq 100$), ktoré udáva počet priestorov.

Nasleduje P opisov priestorov, v ktorých sa na prvom riadku opisu nachádzajú vždy dve medzerou oddelené čísla R a S ($1 \leq R, S \leq 100$) udávajúce rozmery priestoru. Nasleduje R riadkov po S znakoch, pričom znaky sú v, ^, >, <, alebo .

Výstup

Na výstupe je pre každý priestor jediné číslo: minimálny počet šípiek, ktoré musia byť opravené, aby robot nikdy nevyšiel z priestoru, bez ohľadu na to, na ktorej zo šípok začne. Ak takýto stav nie je možné dosiahnuť, vypíšte -1.

Vzorový vstup

```
3
3 4
.^..
.>.>
....
3 3
>.v
...
^.<
```

```
3 3
..^
.>.
v..
```

Vzorový výstup

```
2
0
-1
```

Porucha

Na jednom robotovi sa vyskytla porucha interného mikrofónu, ktorá spôsobuje, že pri transformovaní hovorenej reči do textu sa občas opakujú písmená. Napríklad z povedaného „ahoj“ môže byť výsledný text napríklad „ahhooj“.

Jedným riešením je nechať zákazníka zopakovať rovnaké slovo viac krát a z toho zistiť, čo asi povedal.

Pre každý prepis hovoreného slova je možné robiť iba dve operácie:

- Duplikovať ľubovoľný znak, t.j. vložiť rovnaký znak hneď zaň.

Napr. z „ahhooj“ sa po zdublikovaní prvého znaku stane „aahhooj“

- De-dublikovať už zdvojený znak, t.j. ak máme na ľubovoľnom mieste v reťazci dva po sebe idúce rovnaké znaky, urobíme z nich jeden.

Napr. z „ahhooj“ vieme urobiť „ahhoj“

Okrem týchto operácií nie sú žiadne iné operácie povolené (znaky nevieme ľubovoľne vyrábať, premiestňovať či odstraňovať).

Na zistenie, aké je asi správne slovo, si vytvoríme sadu prepisov opakovane hovoreného slova – napríklad „aaahooj“, „ahoj“ a „ahhoojjjj“. Pomocou uvedených dvoch operácií skúsime dosiahnuť, aby všetky prepisy v sade vyzerali úplne totožne. Keď na to použijeme najmenší možný počet operácií, je veľmi pravdepodobné, že to je to skutočne vyslovené slovo.

Vstup

Prvý riadok vstupu obsahuje počet sád prepisov S ($1 \leq S \leq 100$).

Každá sada má na prvom riadku svojho opisu číslo N ($1 \leq N \leq 100$) reprezentujúce počet prepisov v sade. Nasleduje N riadkov, na každom z nich je zadaný jeden prepis dĺžky 1 až 100 vrátane, obsahujúci len malé písmená anglickej abecedy.

Výstup

Na výstupe je pre každú sadu jediné číslo – minimálny počet operácií potrebný v rámci sady na to, aby všetky prepisy v sade vyzerali rovnako. V prípade, že je nemožné taký stav dosiahnuť, vypíšete -1.

Vzorový vstup

```
4
3
aaahooj
ahoj
ahhoojjjj
1
ahoj
3
aaabcfeg
abbcccefggg
abcegf
2
ahoj
aooj
```

Vzorový výstup

```
8
0
-1
-1
```

Doprezentovať

Roboty majú telo zložené z viacerých segmentov, ktorými vedú kvôli prezentačným (alebo tanečným. . .) účelom rôzne pohybovať.

Pre prezentačný podprogram je stav jednotlivých segmentov zakódovaný ako reťazec núl a jednotiek. Napríklad reťazec 10100 predstavuje zakódovanú polohu dvoch segmentov (každá 1 reprezentuje jeden segment) a poloha príslušnej 1 kóduje polohu segmentu. Detaily fungovania prezentačného podprogramu nie sú známe.

Avšak keď je potrebné na konci dňa prezentáciu ukončiť, musí sa stav segmentov dostať do východzej pozície. To znamená, aby všetky 1 boli napravo. Presúvanie 1 má však svoje pravidlá, aby nedošlo k poškodeniu robota.

Najprv si treba zvoliť niektorú z 1, ktorá má napravo od seba 0, a následne ju posunúť čo najviac napravo, t.j. po najbližšiu 1, alebo na okraj (ak ide o najpraverejšiu 1). Takáto operácia trvá $1 + N$ sekúnd, kde N je počet pozícií o koľko sa presunula zvolená 1. Napr. 10100 je možné zmeniť na 01100 za $1 + 1$, t.j. 2 sekundy, alebo na 10001 za $1 + 2$, t.j. 3 sekundy.

Keďže ukončenie prezentácie môžu zákazníci pozorovať cez výklad, potrebujeme, aby návrat do východzej pozície trval čo najdlhšie

Vstup

Vstup začína prvým riadkom s číslom T ($1 \leq T \leq 100$), ktoré označuje počet rôznych scenárov.

Nasleduje T riadkov, každý s reťazcom S zloženého len z cifier 0 a 1 a dĺžky 1 až 100 000 vrátane.

Výstup

Pre každý scenár vypíšte na samostatný riadok jediné číslo – za aký maximálny čas sa klby dostanú do východzej pozície.

Vzorový vstup

```
4
10100
0000011
1100001
001110100011010
```

Vzorový výstup

```
8
0
10
48
```

Akcia

Na podporu nasadenia robotov v niektorých pobočkách sa banka rozhodla odmeniť zákazníkov, ktorí využili Banko-vklado-mat.

Na konci každého dňa sa zo všetkých vkladov uskutočnených od začiatku akcie vyberie najnižší a najvyšší vklad. Zákazník, ktorý uskutočnil najvyšší vklad, získa odmenu rovnú rozdielu medzi jeho vkladom a najnižším vkladom (ale v centoch...). Oba tieto vklady (najnižší a najvyšší) sú následne vylúčené z akcie.

Nasledujúci deň sa scenár opakuje až do skončenia akcie. Potrebujeme zistiť, koľko bolo vyplatené na odmenách, keď pre každý deň vieme, aké vklady sa uskutočnili.

Vstup

Prvý riadok vstupu obsahuje prirodzené číslo N ($1 \leq N \leq 10$) vyjadrujúce počet pobočiek.

Pre každú pobočku je na prvom riadku počet dní akcie D ($1 \leq D \leq 500$). Nasleduje D riadkov, kde prvé číslo v riadku V ($0 \leq V \leq 750$) značí koľko vkladov sa uskutočnilo v daný deň. Nasleduje medzera a následne V čísiel oddelených medzerou – hodnoty vkladov H ($1 \leq H \leq 2\,000\,000\,000$)

Výstup

Jedno číslo pre každú pobočku – celková suma, ktorá sa počas akcie vyplatila. Predpokladajte, že na konci každého dňa existujú aspoň dva vklady.

Vzorový vstup

```
1
3
5 1 9 2 5 7
2 3 6
0
```

Vzorový výstup

```
16
```

Vysvetlenie: Na konci prvého dňa bola vyplatená odmena 8 centov (9 - 1) a vklady 9 a 1 odstránené. Na konci druhého dňa bola vyplatená odmena 5 centov (7 - 2) a vklady 7 a 2 odstránené (berieme do úvahy všetky neodstránené vklady od začiatku akcie). Na konci posledného dňa bola vyplatená odmena 3 centy (6 - 3) a vklady 6 a 3 odstránené.

Atramentcam

Robot pre svoju činnosť potrebuje zo záberu kamery citlivej na špeciálny atrament (najmä pre účely OCR a detekcie šípiek...) vyhľadať segment, na ktorý sa pri spracovaní obrazu treba bližšie zamerať. Kamera poskytuje snímku obdĺžnikového tvaru, kde každý pixel reprezentuje intenzitu odrazu laseru od špeciálneho atramentu.

Po viacerých experimentoch sa ako najdôležitejšia oblasť javí ľubovoľne veľký (ale zmestiaci sa do záberu) štvorcový výrez, pre ktorý platí, že súčet intenzít v ňom je maximálny možný. Potrebujete zistiť, aký je tento maximálny súčet.

Vstup

Na prvom riadku sa nachádza jedno číslo N ($1 \leq N \leq 10$), ktoré označuje počet snímok z kamier s rôznym rozlíšením.

Každá snímka má v prvom riadku svojho opisu dve čísla R a S ($1 \leq R, S \leq 300$), kde R je počet riadkov obdĺžnika a S je počet stĺpcov. Na nasledujúcich R riadkoch sa nachádza vždy S medzerou oddelených čísiel C ($-1000 < C < 1000$).

Výstup

Na výstupe je pre každú snímku jeden riadok, kde sa nachádza suma čísiel štvorca s maximálnym súčtom.

Vzorový vstup

```
1
10 15
-5 -9 2 -4 -7 7 -6 2 -7 -2 7 9 0 8 7
1 -2 8 1 -3 0 2 -6 9 4 0 2 3 -9 -4
3 4 1 8 4 -1 -9 -3 -6 -2 -1 3 -8 -7 -7
5 -8 4 5 3 9 3 -5 -5 -6 9 7 4 -4 -6
3 -9 4 -4 -5 4 -2 7 -4 2 8 1 -8 1 -7
0 -8 6 9 -2 3 -7 2 7 4 -2 8 7 8 0
-2 8 -7 1 8 -5 3 -1 -3 8 7 -8 6 -4 -9
6 -7 -9 4 1 7 -6 -9 -1 4 -1 -2 -1 6 -9
-9 -4 -4 -2 -5 -6 -1 -2 -9 1 -7 -4 -9 8 3
1 6 6 -5 2 -7 -9 -2 -7 8 4 5 2 -1 -4
```

Vzorový výstup

```
54
```

Vysvetlenie: Štvorec s maximálnym súčtom 54 má dĺžku strany 5 a ľavý horný roh na riadku 4 a stĺpci 10 (je tam číslo -6).

Zvuková vlna

Pre rozpoznanie zvuku musí robot spracovať jednorozmerné pole čísel, ktoré vyjadrujú amplitúdu zvukovej vlny v danom momente. Pri experimentovaní s rozoznávaním slabík ste identifikovali potrebu upravovať a rýchlo vypočítavať špeciálnu funkciu nad poľom reprezentujúcim zvukovú vlnu.

Máme teda dva typy operácií – COUNT a UPDATE. Pre operáciu COUNT spočítame počet súvislých podpostupností medzi X -tým a Y -tým prvkom poľa, ktoré majú súčet prvkov deliteľný zadaným číslom M . Pre operáciu UPDATE nastavíme hodnotu P -teho prvku na novú hodnotu R .

Vstup

Na prvom riadku vstupu je celé číslo T ($1 \leq T \leq 10$), ktoré označuje počet testov na vstupe.

Pre každý test je najprv na samostatnom riadku uvedené celé číslo N ($1 \leq N \leq 10^5$) a M ($2 \leq M \leq 10$) oddelené medzerou, kde N označuje počet prvkov v poli. Na ďalšom riadku sa nachádza N celých čísel oddelených medzerou, kde i -te číslo predstavuje i -ty prvok poľa (každé z týchto čísel sa zmestí do 32-bitovej celočíselnej premennej). Na ďalšom riadku sa nachádza počet operácií Q ($0 \leq Q \leq 10^5$).

Nasleduje Q riadkov, ktoré opisujú jednotlivé operácie. Pre operáciu COUNT je na vstupe uvedený riadok „COUNT X Y “, kde X ($0 \leq X < N$) a Y ($X \leq Y \leq N$) sú celé čísla reprezentujúce interval indexov $\langle X, Y \rangle$. Pre operáciu UPDATE je na vstupe uvedený riadok „UPDATE P R “, kde P ($0 \leq P < N$) je index poľa a R je 32-bitové celé číslo, ktoré predstavuje novú hodnotu P -teho prvku poľa.

Výstup

Pre každú operáciu COUNT vypíšte na samostatný riadok práve jedno číslo – počet súvislých podpostupností v poli, ktoré sa nachádzajú medzi X -tým a Y -tým prvkom poľa, pričom každá takáto podpostupnosť má súčet prvkov deliteľný číslom M .

Vzorový vstup

```
1
5 3
2 1 4 3 2
4
COUNT 2 4
COUNT 1 5
UPDATE 2 2
COUNT 1 5
```

Vzorový výstup

1
2
3

Vysvetlenie vzorového vstupu a výstupu: Pre prvú operáciu COUNT máme spočítať počet súvislých podpostupností v podpostupnosti 4 3. Iba podpostupnosť 3 má súčet prvkov deliteľný tromi.

Pre druhú operáciu COUNT máme spočítať počet súvislých podpostupností v podpostupnosti 1 4 3 2. Podpostupnosti 4 3 2 ($4 + 3 + 2 = 9$) a 3 majú súčet prvkov deliteľný tromi.

Operácia UPDATE mení hodnotu prvku na druhom indexe 4 na novú hodnotu 2. Pre tretiu operáciu COUNT máme spočítať počet súvislých podpostupností v podpostupnosti 1 2 3 2. Podpostupnosti 1 2 ($1 + 2 = 3$), 1 2 3 ($1 + 2 + 3 = 6$) a 3 majú súčet prvkov deliteľný tromi.