

Roboti (roboti)

Mega i Ultra su dva najsavremenija robota koji su došli u obilazak grada Visokog. Kako su tu po prvi put, sa sobom su ponijeli i mapu grada na kojoj se nalaze sve lokacije, ulice, raskrsnice, podzemni prolazi, žičare i sl. Obilazak bi sasvim sigurno bio puno jednostavniji da ne moraju štedjeti baterije koje ih pokreću i koje se, nažalost, veoma brzo troše. Mega je toga naročito svjestan i veliku pažnju posvećuje planiranju obilaska, tako da je potrošnja baterija što manja. Na mapi je između svake dvije povezane lokacije naznačio i količinu energije potrebnu da bi se s jedne lokacije stiglo u drugu. S druge strane, Ultra je veliki obožavatelj teorije brojeva i u slobodno vrijeme istražuje različite načine na koje je moguće otkriti da li je neki broj djeljiv sa tri. Štaviše, njegova ljubav je prerasla u opsesiju tako da on izričito zahtjeva da vrijednost potrošene energije pri prelasku s jednog mjesta na drugo tokom obilaska pri dijeljenju sa 3 daje ostatak 0, 1 ili 2, sve u zavisnosti od njegovog raspoloženja. To dodatno otežava složene proračune i pripreme koje Mega izvodi kako bi maksimalno uštedio baterije, i što je najgore, uzima mu vrijeme koje je planirao provesti u obilasku. Meginu nezavidnu situaciju bi značajno popravio program kojeg bi on mogao iskoristiti u sklopu vlastitog softvera. Taj program bi za određen dio mape grada izračunao broj svih minimalnih potrošnji energije pri prelasku iz jednog mjesta u drugo koje se nalaze unutar određenog intervala, a pri dijeljenju sa 3 daju ostatak 0, 1 ili 2. Na njegovu sreću, u Visokom se našao u pravom trenutku. Program ćete mu napisati vi, a on će za to vrijeme nastaviti sa obilaskom grada, koliko god to bilo teško imajući Ultru pored sebe.

Zadatak

Vaš zadatak je da napišete funkciju *BrojMinimalnihPotrosnji* koja kao argumente prima pet pozitivnih cijelih brojeva r , p , q , v , e , kao i niz pozitivnih cijelih brojeva (veličine $3 \cdot e$). Broj r predstavlja traženi ostatak djeljenja sa 3, dakle, može imati vrijednosti 0, 1 ili 2. Brojevi p i q su donja i gornja granica potrošene energije pri obilasku, respektivno. v predstavlja broj označenih lokacija na mapi, a e broj naznačenih potrošnji energije između povezanih lokacija na mapi. Niz koji se prosljeđuje funkciji u svakom bloku od tri elementa sadrži podatke o početnoj i krajnjoj lokaciji, koje su indeksirane brojevima počevši od 1, kao i o količini energije koja će se potrošiti da bi se došlo iz početne u krajnju lokaciju (ne nužno i obrnuto) sa vrijednostima između 1 i 100, uključujući. Brojevi p i q su uključeni u granice, tako da je p strogo manje od q . Maksimalna vrijednost broja q je 10.000. Moguće je da između nekih lokacija nema naznačenih potrošnji, ali

i s druge strane, da je između neke dvije lokacije naznačeno više od jedne potrošnje.

Podzadatak 1 (18 bodova): $3 \leq v \leq 20$, $1 \leq e \leq 100$

Podzadatak 2 (33 boda): $3 \leq v \leq 100$, $1 \leq e \leq 1.000$

Podzadatak 3 (49 bodova): $3 \leq v \leq 150$, $1 \leq e \leq 2.500$

Primjeri

Primjer 1

BrojMinimalnihPotrosnji (0, 1, 20, 5, 5{1, 2, 12, 2, 3, 3, 3, 4, 3, 4, 1, 4, 1, 5, 5}) = 9

Obrazloženje: Da bi se stiglo od mjesta označenog sa brojem 1 do mjesta označenog sa brojem 2, tako da se pri tome potroši minimalna količina energije, koja u ovom slučaju treba da bude djeljiva sa 3 je 12. Od mjesta 1 do mjesta 3, 15. Od mjesta 1 do mjesta 4, 18. Od mjesta 1 do mjesta 5 27. Od mjesta 2 do mjesta 3, 3. Od mjesta 2 do mjesta 4, 6. Od mjesta 2 do mjesta 5, 15. Od mjesta 3 do mjesta 4, 3. Od mjesta 3 do mjesta 5, 12. I na kraju od mjesta 4 do mjesta 5, 9. Od ovih 10 minimalnih potrošnji, 9 se nalazi u granicama između 1 i 20.

Primjer 2

BrojMinimalnihPotrosnji (1, 10, 50, 3, 5{1, 2, 5, 1, 2, 6, 2, 1, 4, 3, 1, 20, 3, 2, 7}) = 4

Detalji implementacije

Sa servera za takmičenje možete preuzeti pripremljena okruženja (*roboti_c.zip*, *roboti_cpp.zip* ili *roboti_pas.zip*) sa osnovnim fajlovima za C, C++ i Pascal.

Ukoliko koristite C ili C++, napišite funkciju s prototipom:

int BrojMinimalnihPotrosnji(**int** r, **int** p, **int** q, **int** v, **int** e, **int*** energije);
u fajlu *roboti.[c/cpp]*.

Ukoliko koristite Pascal, napišite funkciju sa prototipom:

function BrojMinimalnihPotrosnji (r : **LongInt**; p : **LongInt**; q : **LongInt**;
v : **LongInt**; e : **LongInt**; var energije : **Array of LongInt**) : **LongInt**;
u fajlu *roboti.pas*.

Samo unutar ovog fajla treba da implementirate svoje rješenje. Pri tome smijete koristiti i druge pomoćne funkcije koje ste vi napisali, te standardna zaglavlja/biblioteke odabranog programskog jezika i funkcije iz ovih biblioteka. Ne smijete ni na koji način vršiti interakciju sa standardnim ulazom/izlazom niti sa bilo kojom datotekom.

U pripremljenom okruženju nalazi se fajl *grader.[c/cpp/pas]* koji testira ispravnost rada¹ funkcije koju ste napisali na javne testne primjere. Kada šaljete svoje rješenje, šaljete samo fajl *roboti.[c/cpp/pas]*, dok komisija koristi svoj *grader.[c/cpp/pas]* koji nije javni. U skladu s tim, slobodni ste da modificirate *grader.[c/cpp/pas]* i prilagođavate ga svojim potrebama u svrhu testiranja na lokalnom računaru.

Ukoliko koristite *Code::Blocks* u pripremljenim okruženjima možete naći i odgovarajuće projekte sa podešenim parametrima za prevođenje. *Release build* u potpunosti odgovara parametrima za prevođenje koji su na serveru za takmičenje, dok *Debug build* ima isključene optimizacije i uključene simbole za debugiranje.

Ukoliko koristite *FreePascal IDE*, dovoljno je da pokrenete prevođenje fajla *grader.pas* dok je u istom folderu fajl *roboti.pas*. Na serveru za takmičenje postavljeni su sljedeći parametri za prevođenje: `-dEVAL -vw -XS -O2`.

Ukoliko ne koristite *Code::Blocks*, odnosno *FreePascal IDE*, u okruženjima se nalaze i fajlovi *prevedi_[c/cpp/pas].sh* koje možete koristiti za prevođenje svojih programa, a koje pozivate iz terminala komandom `sh prevedi_[c/cpp/pas].sh` iz odgovarajućeg foldera.

¹ Fajl *grader.[c/cpp/pas]* koji je javno dostupan testira samo ispravnost bez postavljanja ograničenja na vrijeme izvršavanja i iskorištenu memoriju.