

U1. Slaptažodžiai. Parašykite programą, kurį tikrina, ar faile surašyti slaptažodžiai yra „tinkami“ naudojimui ar ne. Slaptažodžiams sudaryti naudojamos didžiosios (ABCDEFGHIJKLMNOPQRSTUVWXYZ) ir mažosios (abcdefghijklmnopqrstuvwxyz) lotyniškos raidės, bei skaitmenys (0123456789). Tinkamas slaptažodis turi tenkinti šias tris sąlygas: 1) Slaptažodis turi būti ne trumpesnis, nei 5 simboliai, bet ne ilgesnis, nei 20 simbolių. 2) Slaptažodis privalo turėti bent vieną didžiąją, bent vieną mažąją raidę ir bent vieną skaitmenį. 3) Slaptažodyje negalimas to paties simbolio naudojimas 3 ir daugiau kartų iš eilės.

Duomenys. Failo U1.txt pirmoje eilutėje yra n ($1 \leq n \leq 50$) slaptažodžių skaičius faile. Kiekvienoje sekančioje eilutėje yra pateikiami slaptažodžiai (viena eilutė – vienas slaptažodis).

Rezultatai. Ekrane kiekvienoje eilutėje spausdinkite žodžių porą: slaptažodį, bei žodžius „TAIP“, jei slaptažodis tinkamas, arba „NE“, jei slaptažodis nėra tinkamas.

U1.txt	Ekranas
7	6oRt NE
6oRt	12345 NE
12345	Blogas NE
Blogas	Gera5 TAIP
Gera5	ilgasBlogas NE
ilgasBlogas	ilgasGera5 TAIP
ilgasGera5	ilgaaasBloga5 NE
ilgaaasBloga5	labaiIlgasNelabaiGera5 NE
labaiIlgasNelabaiGera5	

U2. Laidai. Elektrikas turi sulituoti n laidų į schemą. Schemos litavimo taškai yra sunumeruoti. Kiekvienam laidui nurodomi du taškai, išreikšti taškų numeriais a_i ir b_i , laido varža r_i ir laido kaina k_i . Jeigu lituojant susidaro ciklas, mažiausios varžos laidas sudegą. Jei yra keli vienodos varžos laidai, sudegą anksčiausiai prilituotas laidas. Gali būti daugiau, nei vienas laidas tarp bet kurių dviejų taškų. Parašykite programą, kuri nurodytų, kokia tvarka reiktų lituoti laidus, kad galutinėje schemoje sulituotų ir nesudegusių laidų kaina būtų didžiausia.

Duomenys. Tekstinio failo U2.txt pirmoje eilutėje nurodytas laidų kiekis n ($1 \leq n \leq 3000$). Tolesnėse n eilučių yra po 4 sveikuosius skaičius a_i , b_i , r_i , k_i ($1 \leq a_i, b_i, r_i, k_i \leq 5000$, $a_i \neq b_i$). Laidai numeruojami nuo 1 pagal pateiktą duomenų seką.

Rezultatai. Į ekraną pirmoje eilutėje išveskite maksimalią kainą, antroje – sudegusių laidų numerius pagal litavimo eigą, o trečiojoje – visų laidų numerius pagal litavimo eigą. Jeigu yra keletas variantų, pateikite vieną iš jų.

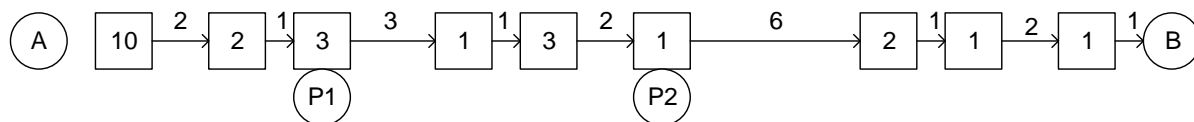
U2.txt	Ekranas	Pastabos
4	5	Svarbu tik, kad 2 būtų prieš 1.
10 20 5 3	2	
20 11 5 2	2 3 1 4	
10 11 7 1		
1 2 1 1		

U3. Transportavimas. Tarp dviejų miestų A ir B užveisti pramoniniai vaismedžių sodai. Miestus jungia kelias, kuriuo vaisiai vežami į supirkimo punktą, esantį mieste B. Dėžės su vaisiais sustatomos pakelėje vienoje eilėje, mašinos surenka dėžes ir jas nuveža į supirkimo punktą. Vaisiai transportuojami tik viena kryptimi: iš A į B. Vienas supirkimo punktas nebespėja priimti visų vaisių, todėl buvo nuspręsta pastatyti dar du supirkimo punktus P1 ir P2 kelyje iš A į B.

Parenkite programą, kuri parinktų punktų P1 ir P2 vietas kelyje taip, kad transportavimo išlaidos būtų minimalios. Vieno kilogramo vaisių pervežimas vieno kilometro atstumu kainuoja vieną piniginių vienetą. Punktų P1 ir P2 priėmimo galia ribojama priimamu bendru dėžių svoriu atitinkamai S1 ir S2. Jei dėžė netelpa į punktą, ji nedalinama, o vežama toliau paskutinė paimta dėžė.

Duomenys. Failo U3.txt pirmoje eilutėje įrašytas vaisių dėžių skaičius n ($2 \leq n \leq 200$) ir punktų bendro dėžių svorio apribojimai s_1 ($0 \leq s_1 \leq 200$) bei s_2 ($0 \leq s_2 \leq 200$). Dėžės numeruojamos iš eilės nuo 1 iki n , einant iš miesto A į miestą B. Tolesnėse n eilučių įrašyta po 2 sveikuosius skaičius: dėžės svoris s ($1 \leq s \leq 100$) ir atstumas kilometrais tarp dviejų gretimų dėžių d ($1 \leq d \leq 10$), einant iš miesto A į miestą B. Paskutinė d reikšmė nurodo atstumą tarp paskutinės dėžės ir mieste B esančio supirkimo punkto (žr. paveikslą). Bendros transportavimo išlaidos pervežant vaisius iš miesto A į miestą B neviršija 250000000 piniginių vienetų.

Rezultatai. Ekrane atspausdinkite atstumą tarp P1 ir P2 punktų, atstumą tarp P2 punkto ir miesto B, minimalias transportavimo išlaidas.



$$10 * (2 + 1 + 3 + 1 + 2 + 6 + 1 + 2 + 1) + 2 * 1 + 1 * (1 + 2) + 3 * 2 + 2 * (1 + 2 + 1) + 1 * (2 + 1) + 1 * 1 = 213$$

U3.txt	Ekranas
9 7 8	6
10 2	10
2 1	213
3 3	
1 1	
3 2	
1 6	
2 1	
1 2	
1 1	