

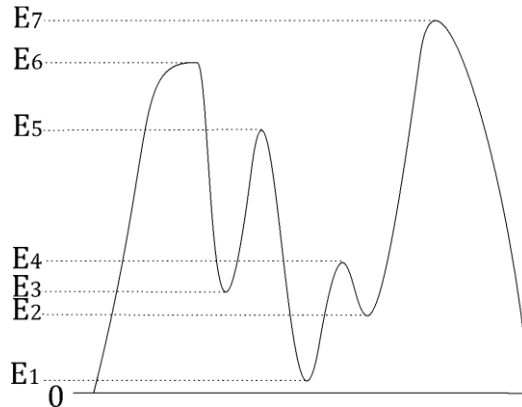
Viršūnės

Kalnuotoje saloje alpinistas užkopė į vieną viršūnę ir nori pasiekti bet kurią kitą aukštesnę viršūnę.

Tiksliau, kiekvienas salos taškas yra tam tikrame *aukštyje* virš jūros lygio (kur aukštis 0). Jei pasiektos viršūnės aukštis yra E_i , tai alpinisto tikslas yra pasiekti kitą viršūnę E_j kur $E_j > E_i$. Kadangi alpinistas yra viršūnėje, tai norėdamas patekti aukščiau, alpinistas visų pirma turi nusileisti žemiau ir tik tada jis gali vėl kilti aukšty. Kelias žemyn niekada nėra toks įspūdingas kaip kelias aukšty, todėl alpinistas nori, kad žemiausias kelio nuo dabartinės iki aukštesnės viršūnės taškas būtų kuo aukščiau.

Paveikslėlyje pateiktame pavyzdyje alpinistas yra viršūnėje, kurios aukštis E_4 . Yra trys aukštesnės viršūnės (E_5 , E_6 ir E_7), tačiau kelias, kurio žemiausias taškas būtų aukščiausiai, yra tik į viršūnę E_7 – reikia nusileisti tik iki aukščio E_2 . Kitais atvejais reikėtų nusileisti iki E_1 . Jei alpinistas pradeda nuo E_5 , žemiausia vieta yra E_3 (pakeliui į E_6), o jei nuo E_6 – E_1 .

Salos žemėlapis yra dvimatė stačiakampė lentelė, sudaryta iš $N \times M$ kvadratinų langelių. Langelyje įrašytas skaičius nusako atitinkamo salos regiono aukštį virš jūros lygio. Du langeliai



yra *gretimi*, jei jie turi bendrą tašką. Taigi, kiekvienas langelis (išskyrus kraštinius) yra gretimas aštuoniems kitiems. *Kelias* yra gretimų langelių seka. *Plokštuma* yra to paties aukščio vieno ar daugiau langelių aibė, kurioje bet kuriuos du langelius jungia kelias sudarytas tik iš tos aibės langelių. Du gretimi vienodo aukščio langeliai visuomet priklauso tai pačiai plokštumai. *Viršūnė* yra plokštuma, kurios langeliai neturi gretimų aukštesnių langelių.

Parašykite programą, kuri surastų visas salos viršūnes ir kiekvienai iš jų rastų didžiausią galimą žemiausio taško pakeliui į aukštesnę viršūnę aukštį. Užkopęs į pačią aukščiausią salos viršūnę, alpinistas paliks šią salą ieškodamas aukštesnių viršūnių kituose kalnuose, todėl žemiausias ieškomas taškas tokiai viršūnei bus jūros lygyje (0).

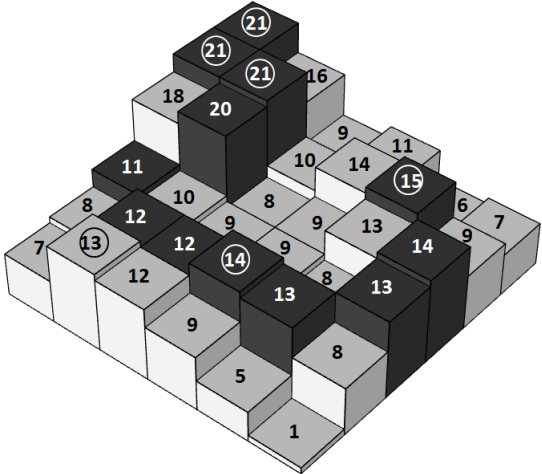
Pradiniai duomenys

Pirmoje failo **peaks.in** eilutėje įrašyti du sveikieji skaičiai N ir M ($1 \leq N, M \leq 2000$; $N \times M \leq 10^5$), žemėlapio aukštis ir plotis, atitinkamai. Tolesnėse N eilučių pateiktas pats salos žemėlapis. Kiekvienoje iš šių eilučių yra M tarpais atskirtų sveikųjų skaičių E_{ij} ($1 \leq E_{ij} \leq 10^6$). Langelio, esančio žemėlapio i -ojoje eilutėje ir j -ajame stulpelyje, aukštis E_{ij} yra j -asis skaičius $i+1$ -ojoje failo eilutėje.

Rezultatai

Pirmoje failo **peaks.out** eilutėje turi būti įrašytas vienas sveikasis skaičius P , viršūnių kiekis saloje. Kiekvienoje tolesnių P eilučių turi būti įrašyti du sveikieji skaičiai: viršūnės aukštis ir didžiausias žemiausio taško pakeliui iš jos į aukštesnę viršūnę aukštis. Viršūnės turi būti pateiktos jų aukščio nedidėjimo tvarka; jei kelios viršūnės yra to paties aukščio, jos pateikiamos žemiausio taško aukščio nedidėjimo tvarka.

1 pavyzdys

Pradiniai duomenys (failas <code>peaks.in</code>)	Rezultatai (failas <code>peaks.out</code>)	Paiškinimas:
6 6 21 16 9 11 6 7 21 21 10 14 15 9 18 20 8 9 13 14 11 10 9 9 8 13 8 12 12 14 13 8 7 13 12 9 5 1	4 21 0 15 11 14 13 13 12	 <p>Visi viršūnėms priklausantys langeliai pažymėti apskritimais. Vienas galimas kelias iš viršūnės, kurios aukštis 15, patamsintas.</p>

2 pavyzdys

Pradiniai duomenys (failas <code>peaks.in</code>)	Rezultatai (failas <code>peaks.out</code>)
5 3 16 14 16 14 14 15 12 17 16 12 13 10 16 11 16	5 17 0 16 15 16 14 16 13 16 13

Vertinimas

Testai, kuriuose $N \leq 2$ arba $M \leq 2$, yra verti 15 taškų.

Testai, kuriuose $P \leq 500$, yra verti 50 taškų.

Testai, kuriuose $P \leq 5000$, yra verti 80 taškų.