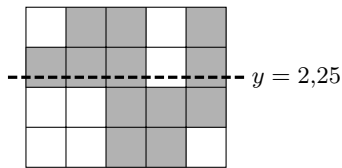


Napiši podprogram (funkcijo), ki pregleda zemljevid kraljestva in vrne y -koordinato tiste vodoravne črte (premice), za katero sta površini kraljestva pod in nad črto enaki. Če ob razrezu kraljestva „podkraljestvi“ razpadeta na več kosov, nas to ne moti. Koordinatni sistem postavimo tako, da ima izhodišče v spodnjem levem vogalu zemljevida, enota pa je enaka stranici enega kvadrata na karirasti mreži. Pozor — iščemo točno vrednost za y in ta ni nujno celoštevilska. (Predpostavi, da so podatkovni tipi, ki jih tvoj programski jezik uporablja za predstavitev realnih števil (npr. **float** ali **double**) dovolj natančni za potrebe te naloge, torej ti ni treba skrbeti zaradi morebitnih majhnih zaokrožitvenih napak.)

Nalogo lahko rešiš tudi tako, da izpišeš *celoštevilski* y , ki čim boljše (čeprav morda ne čisto natančno) razpolovi kraljestvo. Takšne rešitve bodo vredne največ 10 točk (od 20).

Pri pisanju podprograma privzemi, da imaš nekje že definirani spremenljivki w in h , ki hranita širino oz. višino zemljevida (v številu kvadratkov), ter funkcijo $Znotraj(x, y)$, ki vrne 1, če je kvadrata s koordinatama (x, y) na zemljevidu črn, sicer pa 0.

Primer:



Na sliki je $w = 5$, $h = 4$. Vrisana je tudi iskana rešitev — meja poteka na višini $y = 2,25$. V tem primeru je vrstica razrezana po četrtini in zato južni polovici kraljestva prispeva en kvadratak ploščine, severni polovici pa tri kvadratke. Tako imata severna in južna polovica res enako skupno ploščino, namreč 6 kvadratkov. Rešitev za 10 točk, ki vedno vrača celoštevilske y , pa bi morala pri primeru s slike vrniti $y = 2$.

4. Strukturirani podatki

Na vhodni datoteki imamo strukturirano besedilo v obliki, ki je podobna HTML ali XML, a s preprostejšim zapisom. Lahko si ga predstavljamo kot knjigo, ki vsebuje poglavja, ta vsebujejo podpoglavja, ta odstavke in tako naprej.

Vsaka vrstica vhodne datoteke lahko vsebuje začetno značko, končno značko ali pa neko poljubno vrstico besedila.

Začetna značka je sestavljena iz znaka „+“, ki mu sledi neko poljubno ime značke, npr:

+uvod

Končna značka je sestavljena iz znaka „-“, ki mu sledi ime značke, ki se na tem mestu zaključuje, npr:

-uvod

Predpostavimo lahko, da so vhodni podatki pravilni: začetne in končne značke so lahko gnezdene, pri tem se ime končne značke vedno ujema z imenom zadnje začetne značke, zato ga ni treba preverjati.

Vse ostale vrstice lahko vsebujejo poljubno besedilo. Te vrstice se ne morejo začeti z znakoma „+“ ali „-“.

Zaradi preglednosti lahko na začetku vsake vrstice stojijo presledki, ki pa jih ignoriramo.

Primer takega besedila na vhodni datoteki:

```
+ena
  +dve
    alfa
  -dve
+tri
  beta
  gama
+stiri
  delta
  -stiri
  epsilon
-tri
zeta
+tri
-tri
-ena
eta
```

Napiši program, ki bo bral vhodne podatke v tej obliki in sproti izpisoval vse vrstice besedila, ki niso značke; pri tem pa naj pred vsako vrstico izpiše vse trenutno aktivne značke (v takem vrstnem redu, kot so gnezdene druga v drugi), med seboj ločene s pikami. Tvoj program lahko bere s standardnega vhoda ali pa iz datoteke `vhod.txt` (kar ti je lažje). Bere naj vse do konca (EOF). Predpostavi, da je posamezna vrstica dolga največ 100 znakov in da značke niso gnezdene več kot 100 nivojev globoko. Imena značk so sestavljena le iz črk angleške abecede.

Takle naj bo rezultat obdelave zgornjega primera:

```
ena.dve alfa
ena.tri beta
ena.tri gama
ena.tri.stiri delta
ena.tri epsilon
ena.zeta
eta
```

5. Največji pretok

Imamo mrežo enostavnih merilnih računalnikov. Vsakega sestavljajo procesor, senzor in pet komunikacijskih kanalov. Štirje kanali so namenjeni povezavi med računalniki: *L* (levo), *D* (desno), *Z* (zgoraj) in *S* (spodaj). Peta povezava, *M* (merilnik) sprejema podatke od senzorja.