

Qyteti Ideal

Leonardo, si shume shkencetare dhe artiste te tjere italiane te moshes se tij, ishte shume i interesuar ne planifikimin e qytetit dhe dizenjimin urbanistik. Ai kishte si qellim te modelonte qytetin ideal: qetesues, me hapësire dhe racional ne perdorimin e burimeve te tij, larg nga qytetet e ngushta, klaustrofobike te qyteteve te mesjetes.

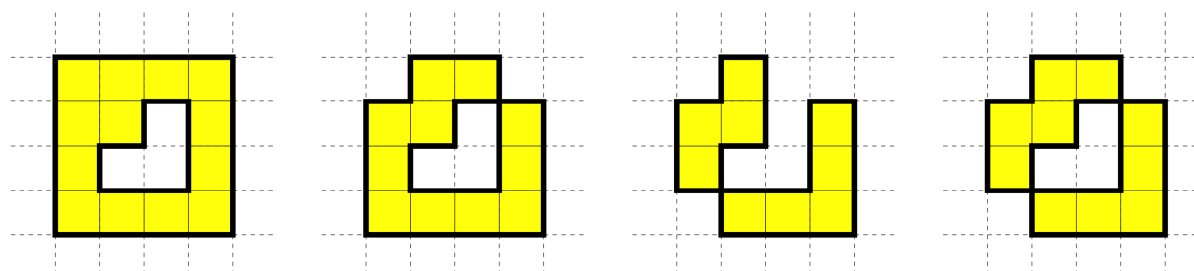
Qyteti ideal

Qyteti perbehet nga N blloqe te vendosura ne nje gride te pafundme me qeliza. Cdo qelize identifikohet nga nje cift koordinatash (rresht, kollone). Per qelizen (i, j) , qelizat fqinje (nese ekzistojne) jane: $(i - 1, j)$, $(i + 1, j)$, $(i, j - 1)$, dhe $(i, j + 1)$. Cdo bllok, kur vendoset ne gride mbulon ekzaktesisht nje qelize. Nje bllok mund te vendoset ne qelizen (i, j) atehere dhe vetem atehere kur $1 \leq i, j \leq 2^{31} - 2$. Ne do te perdorim koordinatat e qelizave per t'iu referuar dhe blloqeve siper tyre. Dy blloqe jane fqinje nese jane vendosur ne qeliza fqinje. Ne nje qytet ideal, te gjitha blloqet e tij jane te lidhura ne nje menyre te tille qe nuk ka “vrime” brenda bordurave te tij, do me thene, qelizat duhet te kenaqin te dyja kushtet e meposhtme.

- Per cdo dy qeliza *boshe*, ekziston te pakten nje sekuence me qeliza fqinje *boshe* qe i lidhin ato.
- Per cdo dy qeliza *jo-boshe*, ekziston te pakten nje sekuence me qeliza *jo-boshe* qe i lidhin ato.

Shembull 1

Asnje nga konfigurimet e meposhtme te blloqeve paraqet nje qytet ideal: dy te parat ne te majte nuk kenaqin kushtin e pare, e treta nuk kenaq kushtin e dyte dhe e katerta nuk kenaq asnje nga kushtet.



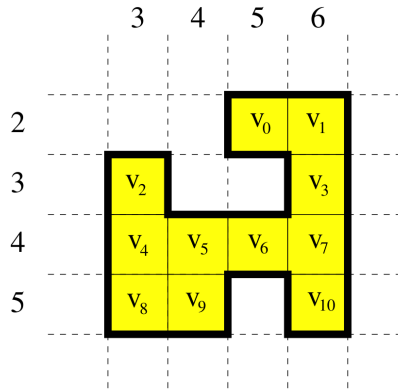
Distanca

Kur pershkohet qyteti, nje kercim tregon kalimin nga nje bllok ne bllokun fqinj. Qelizat boshe nuk mund te pershkohen. Le te jene v_0, v_1, \dots, v_{N-1} koordinatat e N blloqeve qe vendosen ne gride.

Per cdo dy blloqe te dallueshem ne koordinatat v_i dhe v_j , distance e tyre, $d(v_i, v_j)$ eshte numri me i vogel i kercimeve qe kerkohen per te kaluar nga njeri bllok ne tjetrin.

Shembull 2

Konfigurimi i meposhtem paraqet nje qytet ideal te perbere nga $N = 11$ blloqe ne koordinatat $v_0 = (2, 5)$, $v_1 = (2, 6)$, $v_2 = (3, 3)$, $v_3 = (3, 6)$, $v_4 = (4, 3)$, $v_5 = (4, 4)$, $v_6 = (4, 5)$, $v_7 = (4, 6)$, $v_8 = (5, 3)$, $v_9 = (5, 4)$, dhe $v_{10} = (5, 6)$. Per shembull, $d(v_1, v_3) = 1$, $d(v_1, v_8) = 6$, $d(v_6, v_{10}) = 2$, dhe $d(v_9, v_{10}) = 4$.



• Detyra

Detyra juaj eshte te shkruani nje program qe llogarit shumen e distanceve dyshe ndermjet blloqeve v_i dhe v_j per te cilen $i < j$. Formalisht, programi juaj duhet te llogarise vleren e shumes se meposhtme:

$$\sum d(v_i, v_j), \text{ where } 0 \leq i < j \leq N - 1$$

Specifikisht, duhet te implementoni nje rutine(funksion) `DistanceSum(N, X, Y)` qe, pasi eshte dhene N dhe dy tabelat X dhe Y qe pershkruajne qytetin, llogarisin formulen e mesiperme. Te dyja X dhe Y jane te madhesise N ; blloku eshte ne koordinatat $(X[i], Y[i])$ per $0 \leq i \leq N - 1$, dhe $1 \leq X[i], Y[i] \leq 2^{31} - 2$.

Meqenese rezultati mund te jete shume i madh per tu paraqitur duke perdorur 32 bite, duhet ta raportoni ne modul 1 000 000 000 (nje miliard).

Ne shembullin 2, ka $11 \times 10 / 2 = 55$ cifte blloqesh. Shuma e te gjitha distancave cifte eshte 174.

• Nendetyra 1 [11 pike]

Mund te supozoni se $N \leq 200$.

• Nendetyra 2 [21 pike]

Mund te supozoni se $N \leq 2\,000$.

• Nendetyra 3 [23 pike]

Mund te supozoni se $N \leq 100\,000$.

Perveç kesaj, dy kushtet e meposhtme qendrojne: pasi jane dhene dy qeliza jo-boshe i dhe j te tilla qe $X[i] = X[j]$, cdo qelize ndermjet tyre eshte gjithashtu jo-boshe; pasi jane dhene dy qeliza jo-boshe i dhe j te tilla qe $Y[i] = Y[j]$, cdo qelize ndermjet tyre eshte gjithashtu jo-boshe.

- **Nendetyra 4 [45 pike]**

Mund te supozoni se $N \leq 100\,000$.

- **Detajet e implementimit**

Duhet te ngarkoni ekzaktesisht nje skedar, te quajtur `city.c`, `city.cpp` ose `city.pas`. Ky skedar duhet te implementoje subprogramin e pershkruar me siper duke perdorur nenshkrimet vijuese.

Programet C/C++

```
int DistanceSum(int N, int *X, int *Y);
```

Programet Pascal

```
function DistanceSum(N : LongInt; var X, Y : array of LongInt) : LongInt;
```

Keto subprogram duhet te sillen sic eshte pershkruar me siper. Sigurisht qe ju mund te implementoni dhe subprograme te tjera per perdorim te brendshem. Ngarkimet tuaja nuk duhet te bashkeveprojne ne ndonje menyre me input/output-in standart, ose me ndonje skedar tjeter.

Sample grader

Sample grader i siguruar me ambientin e punes do te prese te dhenat ne formatin e meposhtem:

- rreshti 1: N;
- rreshti 2, ..., N + 1: X[i], Y[i].

- **Limitet e kohes dhe memorjes**

- Limiti i kohes: 1 sekond.
- Limiti i memorjes: 256 MiB.