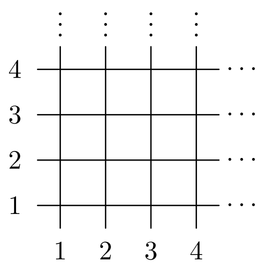
Zbigniew Czech
Treść zadania, Opracowanie

Łukasz Kowalik
Program

OI, Etap II, dzień próbny, 14.02.2006

Magazyn

Ulice w Bajtomieście tworzą prostopadłą siatkę — prowadzą ze wschodu na zachód lub z północy na południe. Ulice północ-południe są ponumerowane od 1 do 500 000 000 w kolejności z zachodu na wschód. Podobnie ulice wschód-zachód są ponumerowane od 1 do 500 000 000 w kolejności z południa na północ. Każda ulica północ-południe przecina każdą ulicę wschód-zachód i odwrotnie, każda ulica wschód-zachód przecina każdą północ-południe. Odległość między dwiema sąsiednimi ulicami północ-południe, a także sąsiednimi ulicami wschód-zachód jest równa jednemu kilometrowi.



W mieście znajduje się k sklepów, a każdy sklep jest położony przy skrzyżowaniu ulic. Kupiec Bajtazar dostarcza towary do każdego z k sklepów, przy czym część sklepów odwiedza kilka razy dziennie. Bajtazar postanowił wybudować magazyn, z którego dostarczałby towary do sklepów. Magazyn powinien być położony przy skrzyżowaniu ulic. Ciężarówka dostarczająca towary w trakcie jednego kursu może odwiedzić tylko jeden sklep — wyjeżdża z magazynu, dostarcza towar do sklepu i wraca do magazynu. Ciężarówka zawsze jedzie najkrótszą trasą z magazynu do sklepu i z powrotem. Odległość między punktami (x_i, y_i) *i* (x_j, y_j) *jest równa*

$$max\{|x_i - x_j|, |y_i - y_j|\}.$$

Zadanie

Napisz program, który:

- *wczyta ze standardowego wejścia opis rozmieszczenia sklepów oraz ile razy dziennie towary są dostarczane do poszczególnych sklepów,*
- *wyznaczy takie położenie magazynu, żeby łączna odległość pokonywana przez ciężarówkę każdego dnia była jak najmniejsza,*
- *wypisze wynik na standardowe wyjście.*

Wejście

Pierwszy wiersz standardowego wejścia zawiera jedną liczbę całkowitą n $(1 \leqslant n \leqslant 100\,000)$*, oznaczającą liczbę sklepów w Bajtomieście.*

Kolejne n wierszy wejścia zawiera opisy sklepów. Wiersz $i+1$-wszy zawiera trzy liczby całkowite x_i, y_i i t_i ($1 \leqslant x_i, y_i \leqslant 500\,000\,000$, $1 \leqslant t_i \leqslant 1\,000\,000$), oddzielone pojedynczymi odstępami. Ten opis oznacza, że i-ty sklep jest położony na skrzyżowaniu x_i-tej ulicy północ-południe i y_i-tej ulicy wschód-zachód i ciężarówka codziennie dojeżdża do tego sklepu t_i razy.

Wyjście

Pierwszy i jedyny wiersz wyjścia powinien zawierać dwie liczby całkowite x_m oraz y_m, oddzielone pojedynczym odstępem i opisujące położenie magazynu jako skrzyżowanie x_m-tej ulicy północ-południe i y_m-tej ulicy wschód-zachód. Jeżeli istnieje wiele poprawnych wyników, Twój program powinien wypisać dowolny z nich.

Przykład

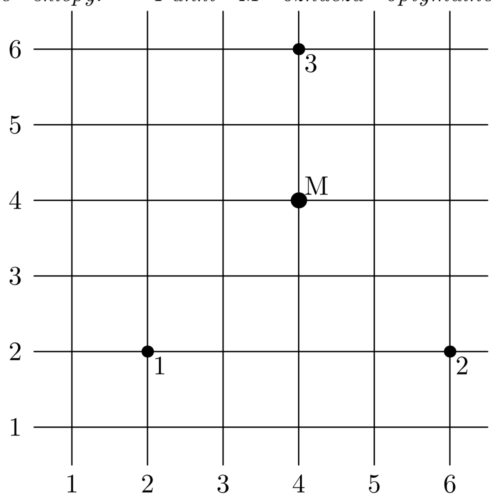
Dla danych wejściowych:

```
3
2 2 1
6 2 1
4 6 1
```

poprawnym wynikiem jest:

```
4 4
```

Poniższy rysunek przedstawia sytuację z przykładowego wejścia. Ponumerowane punkty oznaczają odpowiednie sklepy. Punkt M oznacza optymalne położenie magazynu.



Rozwiązanie

Analiza zadania

Niech $S = \{1,2,\ldots,n\}$ będzie zbiorem sklepów. Niech sklep $i \in S$ ma współrzędne (x_i, y_i) i niech t_i będzie codzienną liczbą pobytów Bajtazara w tym sklepie. Naszym zadaniem jest znalezienie punktu $M = (x_M, y_M)$, dla którego odległość D_m pokonywana przez ciężarówkę każdego dnia jest jak najmniejsza:

$$D_m = \sum_{i \in S} t_i \cdot d_m(M, i),$$

gdzie funkcja $d_m(i, j) = \max\{|x_i - x_j|, |y_i - y_j|\}$ określa odległość między punktami i oraz j.

W ogólnym przypadku, do mierzenia odległości służą funkcje o wartościach nieujemnych zwane metrykami.

Definicja 1 *Metrykę* definiujemy jako dowolną funkcję nieujemną $d(i, j)$, opisującą odległość między punktami i oraz j. Metryka musi mieć następujące własności:

- $d(i,i) = 0$ oraz $d(i, j) > 0$, jeżeli $i \neq j$,
- $d(i, j) = d(j,i)$,
- $d(i, j) \leqslant d(i,k) + d(k, j)$, dla dowolnego punktu k.

Metryka z treści zadania jest *metryką maksimum*. Inną znaną metryką jest *metryka miejska*, zwana też *metryką taksówkową* lub *metryką Manhattan*, zdefiniowana wzorem $d_t(i, j) = |x_i - x_j| + |y_i - y_j|$. Zauważmy, że nasze zadanie byłoby prostsze, gdybyśmy zamiast metryki maksimum mogli użyć metryki miejskiej. Mamy bowiem:

$$D_t = \sum_{i \in S} t_i \cdot d_t(M,i) = \sum_{i \in S} t_i \cdot (|x_M - x_i| + |y_M - y_i|) = \sum_{i \in S} t_i \cdot |x_M - x_i| + \sum_{i \in S} t_i \cdot |y_M - y_i|.$$

Tak więc, aby znaleźć optymalne położenie magazynu, możemy osobno znaleźć współrzędną x_M, dla której suma $\sum_{i \in S} t_i \cdot |x_M - x_i|$ jest minimalna, i osobno współrzędną y_M, dla której minimalna jest suma$\sum_{i \in S} t_i \cdot |y_M - y_i|$. Tym samym zadanie redukuje się do przypadku jednowymiarowego. W dalszym ciągu omówimy metodę rozwiązania przypadku jednowymiarowego dla metryki miejskiej, a następnie pokażemy, jak zastosować ją dla metryki maksimum.

Przypadek jednowymiarowy

Niech $T_L(x)$ (odpowiednio $T_R(x)$) będzie łączną liczbą pobytów Bajtazara we wszystkich sklepach $i \in S$, dla których zachodzi $x_i \leqslant x$ (odpowiednio $x_i > x$) dla pewnej odciętej x, tzn.:

$$T_L(x) = \sum_{i \in S: x_i \leqslant x} t_i \quad \text{oraz} \quad T_R(x) = \sum_{i \in S: x_i > x} t_i.$$

Rozważmy funkcję kosztu $c(x) = \sum_{i \in S} t_i \cdot |x - x_i|$. Łatwo zauważyć, że dla dowolnej odciętej x mamy:

$$c(x+1) = c(x) + T_L(x) - T_R(x).$$

Stąd widać, że wartość funkcji c maleje przy przejściu od x do $x+1$, kiedy $T_L(x) < T_R(x)$, i rośnie, kiedy $T_L(x) > T_R(x)$. Ponieważ funkcja T_L jest niemalejąca, a funkcja T_R — nierosnąca, więc znajdując punkt x, w którym występuje największa „równowaga” pomiędzy wartościami T_L i T_R, znajdziemy miejsce, w którym koszt c jest minimalny. Dokładniej, jeśli znajdziemy odciętą x_i, dla której $T_L(x_i) = T_R(x_i)$, to funkcja c przyjmuje minimalną wartość we wszystkich punktach przedziału $[x_i, x_j]$, gdzie j jest kolejnym po i sklepem w posortowanym ciągu. Jeśli T_L i T_R nie są sobie równe w żadnym punkcie, a x_i jest minimalną wartością, dla której $T_L(x_i) > T_R(x_i)$, to x_i jest jedyną odciętą, dla której koszt c jest minimalny.

Powyższe spostrzeżenia pozwalają nam rozwiązać przypadek jednowymiarowy w osi x (i podobnie w osi y) w czasie $O(n \log n)$, stosując szybki algorytm sortowania, np. sortowanie przez scalanie lub kopcowanie. Alternatywnie można zastosować algorytmy wyznaczające medianę zbioru liczb, np. algorytm Hoare'a — można w ten sposób uzyskać algorytm o złożoności $O(n)$.

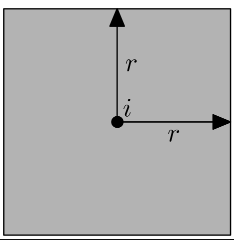
Przypadek dwuwymiarowy

Skonstruujmy rozwiązanie dla przypadku dwuwymiarowego (nadal dla metryki miejskiej) z rozwiązań obliczonych oddzielnie dla obu osi. Jak zauważyliśmy, rozwiązaniem przypadku jednowymiarowego jest albo jeden punkt, albo cały przedział punktów. Zbiór optymalnych rozwiązań przypadku dwuwymiarowego to iloczyn kartezjański rozwiązań przypadków jednowymiarowych. Ostatecznie w wyniku otrzymujemy więc albo pojedynczy punkt, albo odcinek, albo prostokąt złożony z rozwiązań optymalnych.

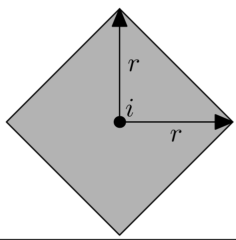
Zmiana metryki

Definicja 2 Dla danej metryki d, punktu i oraz nieujemnej liczby r, definiujemy *kulę domkniętą* $B_r(i)$ jako zbiór punktów, znajdujących się w odległości co najwyżej r od punktu i. Punkt i nazywamy *środkiem*, a r *promieniem* kuli $B_r(i)$.

Kula domknięta $B_r(i)$ w metryce maksimum to kwadrat o środku w punkcie i oraz długości boku $2r$ (rys. 1). Kula domknięta $B_r(i)$ w metryce miejskiej to także kwadrat o środku w punkcie i, tyle że obrócony o 45° i o długości przekątnej $2r$ (rys. 2). Boki obróconego kwadratu mają długość $\frac{2r}{\sqrt{2}}$.



Rys. 1: Kula w metryce maksimum.



Rys. 2: Kula w metryce miejskiej.

Można zauważyć, że kula w metryce maksimum, obrócona o 45° i przeskalowana przez współczynnik $\frac{1}{\sqrt{2}}$ daje kulę o tym samym środku i promieniu, ale w metryce miejskiej. To spostrzeżenie jest pożyteczne wobec prawdziwości następującego twierdzenia:

Twierdzenie 1 *Niech d_1 i d_2 będą metrykami. Jeżeli dla każdego punktu i oraz promienia r, kula o promieniu r oraz środku w punkcie i w metryce d_1 jest taka sama, jak kula o promieniu r oraz środku w punkcie i w metryce d_2, to te dwie metryki są takie same. Oznacza to, że $d_1(i,j) = d_2(i,j)$ dla każdej pary punktów i, j.*

Dowód Niech $B_r^1(i)$ (odpowiednio $B_r^2(i)$) oznacza kulę o promieniu r oraz środku w punkcie i w metryce d_1 (odpowiednio d_2). Niech i, j będą dowolnymi punktami. Przyjmijmy $r = d_1(i, j)$. Wówczas $j \in B_r^2(i)$, a stąd $d_2(i, j) \leqslant r$. Załóżmy, że $d_2(i, j) < r$. Wówczas zachodzi $j \in B_{d_2(i,j)}^1(i)$. To implikuje, że $r = d_1(i, j) \leqslant d_2(i, j) < r$, co daje sprzeczność. Stąd ostatecznie $d_2(i, j) = r$. ■

Tak więc zamiast używać metryki maksimum dla danego zbioru punktów S możemy dokonać obrotu tych punktów o kąt 45°, przeskalować przez współczynnik $\frac{1}{\sqrt{2}}$ i używać metryki miejskiej. Z powyższego twierdzenia wynika poprawność tego postępowania; aby je lepiej zrozumieć, przeprowadzimy jeszcze dowód niekorzystający z ogólnych własności kul w różnych metrykach. Dowolny punkt o współrzędnych (x, y) obrócony o kąt 45° wokół punktu $(0, 0)$ ma współrzędne $(\frac{x-y}{\sqrt{2}}, \frac{x+y}{\sqrt{2}})$. Jeżeli dodatkowo przeskalujemy współrzędne przez współczynnik $\frac{1}{\sqrt{2}}$, to otrzymamy punkt $(\frac{x-y}{2}, \frac{x+y}{2})$. Niech $A = (x_A, y_A)$ i $B = (x_B, y_B)$ będą dowolnymi punktami. Oznaczmy $a = x_A - x_B$ oraz $b = y_A - y_B$. Po obrocie i przeskalowaniu otrzymujemy punkty $A' = (\frac{x_A - y_A}{2}, \frac{x_A + y_A}{2})$ i $B' = (\frac{x_B - y_B}{2}, \frac{x_B + y_B}{2})$. Wówczas odległość w metryce miejskiej jest równa

$$d_t(A', B') = \left|\frac{x_A - y_A - x_B + y_B}{2}\right| + \left|\frac{x_A + y_A - x_B - y_B}{2}\right| = \frac{|a-b| + |a+b|}{2}.$$

Bez straty ogólności możemy przyjąć, że $a \geqslant 0$ (w przeciwnym razie punkty możemy zamienić). Załóżmy, że $|a| \geqslant |b|$. Wówczas $d_m(A, B) = a$ i $d_t(A', B') = \frac{a-b+a+b}{2} = a$. Stąd $d_m(A, B) = d_t(A', B')$. Załóżmy teraz, że $|a| < |b|$. Wówczas $d_m(A, B) = |b|$. Jeżeli $b \geqslant 0$, to $d_t(A', B') = \frac{-a+b+a+b}{2} = b = |b|$. Jeżeli zaś $b < 0$, to $d_t(A', B') = \frac{a-b-a-b}{2} = -b = |b|$. Ponownie mamy $d_m(A, B) = d_t(A', B')$.

Rozwiązania zadania

Rozwiązanie wzorcowe

Opierając się na powyższych rozważaniach można zaproponować następujące rozwiązanie zadania. Najpierw obracamy i przeskalowujemy współrzędne wszystkich sklepów. Zamiast używać współczynnika przeskalowania $\frac{1}{\sqrt{2}}$ używamy współczynnika $\frac{2}{\sqrt{2}}$. Wówczas punkt o współrzędnych (x, y) zostaje przekształcony do punktu o współrzędnych $(x - y, x + y)$. Otrzymane współrzędne są liczbami całkowitymi, co upraszcza obliczenia. Dwa punkty położone w odległości d od siebie w metryce maksimum, po przekształceniu znajdują się w odległości $2d$ w metryce miejskiej. Mimo że wszystkie odległości zostały podwojone, optymalne położenie magazynu w metryce miejskiej odpowiada optymalnemu położeniu w metryce maksimum.

Po wykonaniu przekształcenia otrzymujemy jak gdyby nowe miasto, w którym odległość między dowolnymi dwoma równoległymi sąsiednimi ulicami jest równa dwa kilometry. Nazwijmy to miasto Nowym Bajtomiastem. W Nowym Bajtomieście znajdujemy najlepsze położenie magazynu w metryce miejskiej.

Po uzyskaniu optymalnego położenia magazynu (x, y) w Nowym Bajtomieście, przekształcamy je do punktu o współrzędnych $(\frac{x+y}{2}, \frac{y-x}{2})$ w Bajtomieście. Jeżeli współrzędne x i y są obie parzyste lub obie nieparzyste, to otrzymujemy punkt położony na skrzyżowaniu w Bajtomieście będący poszukiwanym rozwiązaniem. W przeciwnym razie otrzymujemy punkt w środku kwadratu utworzonego przez 4 ulice i 4 skrzyżowania. W takim przypadku jako rozwiązanie zadania wybieramy najlepsze z tych czterech skrzyżowań. Dokonując wyboru nie trzeba liczyć kosztu dostarczenia towaru dla magazynu położonego w każdym z tych czterech skrzyżowań. Wystarczy obliczyć wzrost kosztu dostawy zakładając przemieszczenie magazynu z punktu $(\frac{x+y}{2}, \frac{y-x}{2})$ do każdego z tych skrzyżowań.

Uzasadnimy, że takie postępowanie jest poprawne. Oczywiście nasze cztery skrzyżowania w Bajtomieście odpowiadają czterem skrzyżowaniom, sąsiadującym z punktem (x,y) w Nowym Bajtomieście, tzn. skrzyżowaniom $(x+1,y),(x,y+1),(x-1,y),(x,y-1)$. Jeżeli zbiór optymalnych położeń magazynu w Nowym Bajtomieście nie jest pojedynczym punktem, to jedno z tych sąsiednich skrzyżowań jest także optymalnym położeniem magazynu. Tym samym odpowiadające mu skrzyżowanie w Bajtomieście jest optymalne. Jeśli optymalne położenie magazynu w Nowym Bajtomieście, $s_0 = (x,y)$, jest pojedynczym punktem, to oczywiście, aby uzyskać równie dobry koszt rozwiązania w Bajtomieście, musielibyśmy postawić magazyn pomiędzy sąsiednimi ulicami — tego nam jednak nie wolno. Pokażemy, że wybrana lokalizacja odpowiada drugiemu co do jakości rozwiązaniu z Nowego Bajtomiasta, a więc najlepszemu możliwemu w Bajtomieście. Wystarczy pokazać, że ten „vicelider" jest wśród skrzyżowań $(x+1,y),(x,y+1),(x-1,y),(x,y-1)$.

Załóżmy, że s jest poszukiwanym położeniem vicelidera (czyli jest pewnym skrzyżowaniem w Nowym Bajtomieście). Oczywiście najkrótsza droga z s_0 do s przechodzi przez jedno z czterech sąsiadujących z s_0 skrzyżowań. Przemieszczenie magazynu od s wzdłuż tej drogi w kierunku s_0 powoduje, że łączny koszt dostarczenia towarów pozostaje ten sam lub maleje (wynika to stąd, że w ten sposób w każdym wymiarze nie oddalamy się od położenia s_0). A zatem przyjmując jako nowe położenie magazynu odpowiedniego sąsiada s_0 zamiast s uzyskujemy rozwiązanie niegorsze niż w przypadku s.

Implementacja rozwiązania wzorcowego znajduje się na dysku dołączonym do książeczki w plikach `mag.cpp`, `mag1.cpp` oraz `mag2.pas`.

Inne rozwiązania

Jednym z prostych sposobów rozwiązania zadania jest obliczenie kosztu dostarczenia towaru dla wszystkich możliwych położeń magazynu i wybranie najlepszego położenia. Program działający według tego sposobu zawarty jest w pliku `mags0.cpp`.

Inna możliwość wynika z obserwacji, iż funkcja kosztu osiąga w punkcie optymalnym swoje minima zarówno w osi x, jak i y. Wybieramy więc dowolnie punkt siatki s jako położenie magazynu i obliczamy dla niego wartość funkcji kosztu. Następnie obliczamy koszty dla wszystkich skrzyżowań sąsiednich z punktem s, włączając skrzyżowania sąsiadujące z punktem s po przekątnych. Jeśli istnieje sąsiednie skrzyżowanie o mniejszej w stosunku do s wartości kosztu, to przemieszczamy magazyn do tego skrzyżowania (nowe s) i kontynuujemy postępowanie. Jeśli skrzyżowanie sąsiednie o mniejszej wartości kosztu nie istnieje, to znaczy, że osiągnęliśmy optimum. Program skonstruowany zgodnie z takim postępowaniem zawarty jest w pliku `mags1.cpp`.

Rozwiązania błędne

Jak wskazano powyżej, jednym z możliwych rozwiązań jest poszukiwanie optimum przez badanie sąsiadów bieżącego punktu s i przemieszczanie magazynu do punktu sąsiedniego o najmniejszym koszcie. W pliku `magb0.cpp` zawarty jest program, w którym uwzględnia się jedynie sąsiadów leżących wzdłuż osi x i y, tzn. pomija się sąsiadów leżących po przekątnych. Prowadzi to do błędnych wyników dla niektórych danych testowych.

Źródłem złych rozwiązań mogą być także błędy arytmetyki zmiennopozycyjnej, jeśli stosuje się typy `float` lub `real` (plik `magb1.cpp`).

Testy

Zadanie testowane było na zestawie 15 danych testowych.

Nazwa	n	Opis
mag1.in	3	prosty test poprawnościowy
mag2.in	5	prosty test sprawdzający poprawność
mag3.in	10	współrzędne sklepów to (x,x) dla $x = 1, \ldots, 10$
mag4.in	100	wszystkie sklepy położone w dwóch rogach kwadratu
mag5.in	100	test losowy, sklepy położone na brzegach
mag6.in	100	test losowy
mag7.in	100	test losowy, sklepy położone na brzegach
mag8.in	200	test losowy
mag9.in	30 000	test losowy, sklepy położone na brzegach
mag10.in	70 000	test losowy, sklepy położone na brzegach
mag11.in	70 000	test losowy
mag12.in	90 000	test losowy
mag13.in	100 000	test losowy
mag14.in	100 000	test losowy
mag15.in	100 000	test losowy