



Zadanie: CIE

Ciepło-zimno [B]

Potyczki Algoritmiczne 2016, runda 3. Dostępna pamięć: 256 MB.

23.11.2016

Mała Krotka i jej brat Entek są szczęśliwymi mieszkańcami pewnego d -wymiarowego świata. Dzisiaj postanowili pobawić się w chowanego – jako pierwszy będzie szukał Entek. Ponieważ szukanie w wysokowymiarowych światach nie należy do najłatwiejszych zadań, dla ułatwienia postanowili porozumiewać się przez krótkofalówki. Poza tym każde z nich zabrało ze sobą odbiornik GPS.

Krotka ukryła się w pewnym punkcie kratowym (czyli takim, którego wszystkie współrzędne są całkowite) Hipersześciennego Lasu i nie zamierza się z niego ruszać aż do końca zabawy. Las, zgodnie z nazwą, jest hipersześcianem o boku r – należą do niego punkty o wszystkich współrzędnych z przedziału $[0, r]$. Entek w poszukiwaniu siostry porusza się po lesie, od czasu do czasu podając jej swoją lokalizację odczytaną z GPS-u. Przyjmujemy, że Entek w momencie podawania lokalizacji zawsze znajduje się w punkcie kratowym.

Po podaniu swojej lokalizacji, Entek dostaje od Krotki komunikat „ciepło” albo „zimno” – Krotka informuje go, czy przybliżył się do jej pozycji od miejsca poprzedniego kontaktu.

Dla danych d -wymiarowych punktów p, x, y , Krotka uważa, że punkt x jest bliżej punktu p niż punkt y wtedy i tylko wtedy, gdy

$$\max_{i=1,2,\dots,d} |x_i - p_i| < \max_{i=1,2,\dots,d} |y_i - p_i|.$$

Krótkofalówka Entka ma mało pojemny akumulator i pozwoli mu na nawiązanie co najwyżej k połączeń. Pomóż mu odnaleźć siostrę, zanim straci z nią możliwość kontaktu.

Komunikacja

Krotka ukrywa się w d -wymiarowym punkcie o wszystkich współrzędnych całkowitych i należących do przedziału $[0, r]$. Podczas zabawy Krotka udziela odpowiedzi zgodnie z prawdą i nie porusza się.

Twój program powinien korzystać z dostarczonej biblioteki symulującej odpowiedzi Krotki. Aby skorzystać z biblioteki w rozwiązaniu napisanym w C albo C++, należy wpisać w swoim programie:

```
#include "cielib.h"
```

W przypadku rozwiązań w Javie biblioteka zostanie dołączona bez potrzeby umieszczania dodatkowych linii w pliku źródłowym.

Komunikacja z biblioteką odbywa się za pomocą następujących funkcji:

- `int podajD()` – zwraca wymiar świata, w którym żyją Krotka i Entek, wyżej oznaczony przez d .
- `int podajK()` – zwraca liczbę możliwych pytań `czyCieplo`, wyżej oznaczoną przez k .
- `int podajR()` – zwraca długość boku Hipersześciennego Lasu, wyżej oznaczoną przez r .
- `int czyCieplo(int pozycja[])` – `pozycja` musi być d -elementową* tablicą liczb całkowitych z przedziału $[0, r]$, oznaczającą aktualną pozycję Entka. Ta funkcja zawsze zwraca 0 lub 1. Podczas działania programu można wywołać ją co najwyżej k razy. Przy pierwszym wywołaniu funkcja zwraca wartość 0. Przy każdym kolejnym wywołaniu zwraca wartość 1 wtedy i tylko wtedy, gdy aktualna pozycja Entka jest bliższa pozycji Krotki niż jego pozycja przy poprzednim wywołaniu.
- `void znalazlem(int pozycja[])` – `pozycja` musi być d -elementową tablicą liczb całkowitych z przedziału $[0, r]$ oznaczającą odnaniezoną pozycję Krotki. Funkcja ta musi zostać wywołana dokładnie raz, a jej wywołanie zakończy działanie Twojego programu.

W przypadku języków C i C++ są to funkcje globalne, w przypadku Javy są to statyczne metody klasy `cielib` (czyli wywołuje się je np.: `cielib.podajD()`).

Twój program nie może otwierać żadnych plików ani używać standardowego wejścia i wyjścia. Rozwiązanie będzie kompilowane wraz z biblioteką następującymi poleceniami:

- C: `gcc -std=c11 -static -O2 -s cielib.c cie.c -lm`
- C++: `g++ -std=c++11 -static -O2 -s cielib.c cie.cpp -lm`
- Java:
`javac cie.java`
`jar cf cie.jar *.class`

W dziale „Pliki” możesz znaleźć przykładowe pliki biblioteki i błędne rozwiązania ilustrujące sposób ich użycia. Aby podane powyżej polecenia kompilacji działały, pliki bibliotek powinny znajdować się w bieżącym katalogu.

*W przypadku języków C i C++, przez d -elementową tablicę rozumiemy tablicę co najmniej d -elementową.

Ograniczenia

We wszystkich testach zachodzić będzie $2 \leq r \leq 10^9$, $1 \leq d \leq 500$ oraz $k \geq 100d$.

Ocenianie

Twój program otrzyma punkt za dany test, jeżeli po co najwyżej k wywołaniach funkcji `czyCieplo` wywoła funkcję `znalazlem` z poprawną pozycją Krotki. Program błędnie używający biblioteki (wywołujący funkcje z argumentami nie spełniającymi wymagań podanych w sekcji Komunikacja lub używający standardowego wejścia lub wyjścia) otrzyma za dany test 0 punktów.

Przykładowy przebieg programu

Wywołanie funkcji	Zwracane wartości
<code>podajD()</code>	2
<code>podajK()</code>	200
<code>podajR()</code>	2
<code>czyCieplo({0, 0})</code>	0
<code>czyCieplo({1, 1})</code>	1
<code>czyCieplo({2, 2})</code>	1
<code>znalazlem({2, 2})</code>	Program kończy się sukcesem.