

Задача 3. Камък 0.1 с.  256 MB

Вие сте Боби и обичате лъскави неща. След като намерихте минималното покриващо дърво в графа Ви, под дървото намерихте един лъскав камък, който решихте да потъркате, защото сте Боби и обичате лъскави неща. Камъкът, обаче, беше лъскав с причина и успя да Ви запрати в N -измерна вселена. Вие започвате да огладнявате, затова искате да се върнете във Вашата вселена на произход възможно най-скоро, и за Ваше щастие в една от точките в този N -измерен свят има портал, който ще Ви върне обратно. За Ваше нещастие, не знаете къде е. Разполагате с дрон, който може да се мести в пространството. Той използва един процент от батерията си за всяко преместване, но, тъй като е стар, се зарежда само до 12 процента. Вие го получавате напълно зареден според възможностите му. Можете да го зареждате по всяко време, но за това трябва той да е в точката, в която сте и Вие. Вие и дронът започвате пътешествието в една точка. Вие ставате все по-гладен и по-гладен, затова трябва да мислите бързо как да се измъкнете от N -измерната вселена.

Детайли по имплементацията

Тази задача е интерактивна. Функцията `void solve(long long n)`, която трябва да напишете, ще бъде извикана само веднъж от програмата на журито и като аргумент ще получи цялото число N .

`void solve(long long N, long long M);` - функцията, която трябва да имплементирате. Програмата на журито я вика с параметър N – брой на измеренията и M - максималната позиция. До края на функцията трябва да сте извикали функцията `portal()`.
`std::vector<long long> query();` - Можете да извикате тази функция. Тя връща вектор от N елемента, всеки описващ положението на портала спрямо позицията на дрона. Ако елемент i на вектора е равен на 1, координата в i -тото измерение на портала е по-голяма от координата в i -тото измерение на дрона. Съответно, ако е равен на 0, тя е по-малка, а ако е 2, е равна.

`void moveself(std::vector<long long> pos);` - Можете да извикате тази функция. Тя мести Вас на позицията, описана във вектора `pos`.

`void movedrone(std::vector<long long> pos);` - Можете да извикате тази функция. Тя мести дрона на позицията, описана във вектора `pos`. Не забравяйте, че едно извикване на тази функция Ви коства един процент от батерията на дрона.

`void chargedrone();` - Можете да извикате тази функция. Тя зарежда дрона напълно, т.е. до 12 процента. Ако Вие и дронът не сте в една и съща точка, програмата приключва и ще получите `Wrong Answer`.

`void portal();` - Можете да извикате тази функция. Тя прекратява програмата Ви. Трябва да бъде извикана в края на функцията `solve()`.

`std::vector<long long> getposself();` - Можете да извикате тази функция. Тя връща текущата Ви позиция.

`std::vector<long long> getposdrone();` - Можете да извикате тази функция. Тя връща текущата позиция на дрона.

`void changename(string name);` - Можете да извикате тази функция. Тя променя името Ви на `name`.

`void hugdrone();` - Можете да извикате тази функция. Чрез нея прегръщате дрона, защото сте гладен и Ви е тъжно, когато сте гладен, съответно искате да прегърнете някого (или нещо). Ако Вие и дронът не сте в една точка, програмата приключва и получавате `Wrong Answer`.

Вашата програма `stone.cpp` трябва да имплементира функцията `solve`. Тя може да съдържа и друг код, и функции, необходими за работата Ви, но не трябва да съдържа главната функция `main`. Също така, не трябва да четете от стандартния вход или да отпечатвате на стандартния изход. Програмата Ви трябва да включва хедър файла `stone.h` чрез указание към предпроцесора:

```
#include "stone.h"
```

Ограничения

- $1 \leq N \leq 10^5$

Подзадачи

Подзадача	Точки	Ограничения
1	25	$1 \leq N \leq 1000$
2	75	$1 \leq N \leq 10^5$

Точките за дадена подзадача се получават само ако се преминат успешно всички тестове, предвидени за нея.

Оценяване

Ако сте използвали повече от оптималния брой заявки, получавате 50% от точките на съответния тест.

Локално тестване

Предоставен Ви е файлът `Lgrader.cpp`, който може да компилирате заедно с вашата програма, за да я тествате. При стартиране програмата ще чете от стандартния вход стойностите на N и M . След това ще се отпечата комуникацията, която се извършва. Може да модифицирате предоставения файл, както искате.