



# EL HALCÓN MILENARIO

locked

by Telecode\_2017

Problem

Submissions

Leaderboard

Problema creado por Josean.

El Halcón Milenario es, como todos sabemos, el montón de chatarra más rápido de la galaxia. Puede superar en 5 puntos la velocidad de la luz, y fue la única nave capaz de completar la carrera Kessel en menos de 12 parasegundos. Recientemente, el Halcón ha llegado a manos de la organización del Telecode, que ha tenido ocasión de examinarlo y descubrir cómo puede lograr tan buenos tiempos, y no es debido a motores extremadamente potentes ni fruto de ningún avance en la ingeniería (es más, usa un simple motor estelar turbodiesel de 90CV), sino gracias a una genialidad de las matemáticas.

El campo hiperespacial que general el Halcón Milenario al saltar a la velocidad de la luz se puede modelar, de forma simplificada, como un campo electrostático de módulo  $E$ , que apunta en la dirección positiva del eje  $X$ . Dentro del campo hiperespacial, el Halcón puede modelarse como una pequeña esfera, cargada eléctricamente con una carga  $Q$ , que se mueve debido a la fuerza ejercida por el campo eléctrico sobre ella.

Según estos postulados, el Halcón solo podría dirigirse hacia puntos del espacio que tengan su misma coordenada  $Y$ , dado que el campo eléctrico del hiperespacio no tiene ninguna componente  $Y$ . Esto se soluciona gracias a la avanzada tecnología de túneles espaciales, que podríamos modelar en nuestro hiperespacio simplificado como unas "rampas", sobre las cuales desliza (sin rozamiento) la esfera que representa al Halcón. De esta forma, si el Halcón quisiera ir del punto  $A = (x = 0, y = 0)$  al punto  $B = (x = 3, y = 3)$ , únicamente tendría que generar una rampa entre estos dos puntos, y la fuerza del campo eléctrico le haría rodar sobre la rampa hasta su destino.

Estas rampas hiperespaciales pueden tener diversas formas. Por ejemplo, para llegar del punto  $A = (x = 0, y = 0)$  al punto  $B = (x = 3, y = 3)$ , podríamos utilizar una rampa hiperespacial en forma de línea recta que uniese estos dos puntos. Una nave de masa  $M$  y carga hiperespacial  $Q > 0$  que fuese siguiendo dicha rampa, tardaría un tiempo  $t = \sqrt{(6 \cdot \sqrt{2} \cdot M) / (Q \cdot E \cdot \sin(\pi/4))}$  en llegar a su destino. Una forma muy simple de reducir este tiempo (que es la que usan la inmensa mayoría de cruceros imperiales) es aumentar la carga  $Q$  asociada a la nave, que se hace instalando motores mucho más potentes, lo cual es costoso y muy ineficiente, dado que la curva que relaciona la potencia consumida por el motor con la carga hiperespacial  $Q$  que generan es logarítmica, es decir, requieren mucha más potencia para generar aumentos minúsculos en  $Q$ .

El método que utiliza el Halcón para reducir su tiempo de viaje no consiste en aumentar  $Q$ , sino en seguir una rampa hiperespacial que no tiene forma de línea recta entre el origen y el destino, sino una forma mucho más eficiente, que aunque repercute en una distancia de trayecto más larga, se recorre en un tiempo menor que la línea recta bajo las mismas condiciones de  $Q$ ,  $E$  y  $M$ . Este reto consiste en descubrir cuál es ese camino que sigue el Halcón hasta su destino, y cuál es la longitud del trayecto desde el origen hasta el destino (es decir, la longitud de la rampa hiperespacial entre el origen y el destino).

Para simplificar los casos, asumiremos que el Halcón va desde el punto origen  $A = (x_0, y_0)$ , hasta el punto destino  $B = (x_f, y_f)$ , cumpliéndose siempre  $x_f \geq x_0$  y que  $Q > 0$ . El Halcón sigue siempre la ruta más rápida posible entre el origen y el destino.

## Input Format

Para cada test, la primera línea será la coordenada  $X$  del punto origen, la segunda línea será la coordenada  $Y$  del punto origen, la tercera línea la coordenada  $X$  del punto destino y la cuarta línea la coordenada  $Y$  del punto destino.

## Constraints

$-1000 < x_0, x_f, y_0, y_f < 1000$

## Output Format

El output del programa debe ser una única línea, que contenga la longitud del trayecto que sigue el Halcón entre el origen y el destino, truncado a 5 decimales (no redondeado). El cálculo se debe hacer con la máxima precisión y luego ejecutar el truncado, limitar la precisión puede dar lugar a soluciones erróneas.

## Sample Input 0

0.0  
3.5

1.5  
2.5

**Sample Output 0**

1.84376

**Sample Input 1**

0.0  
3.5  
1.5  
3.5

**Sample Output 1**

1.5

[f](#) [t](#) [in](#)

**Submissions:** [2](#)

**Max Score:** 5000

**Difficulty:** Expert

**Rate This Challenge:**

☆☆☆☆☆

[More](#)

**Admin Options**

[Edit Challenge](#)

[View Submissions](#)

**Current Buffer** (saved locally, editable) [?](#) [↺](#)

BASH ⌵ ⌵ ⚙

1

[Upload Code as File](#)

☐ Test against custom input

Run Code

Submit Code

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)