

Novática, revista fundada en 1975, es el órgano oficial de expresión y formación continua de ATI (Asociación de Técnicos de Informática)

ATI es miembro de CEPIS (Council of European Professional Informatics Societies) y tiene un acuerdo de colaboración con ACM (Association for Computing Machinery). Tiene asimismo acuerdos de vinculación o colaboración con AdaSpain, AI² y ASTIC

<<http://www.ati.es/novatica/>>

CONSEJO EDITORIAL

Antoni Carbonell Noguera, Francisco López Crespo, Julián Marcelo Chocho, Celestino Martín Alonso, Josep Molas i Bertrán, Roberto Moya Quiles, Gloria Nistal Rosique (Presidenta del Consejo), César Pérez Chirinos, Mario Piattini Velthuis, Fernando Piera Gómez, Miquel Sàrries Griñó, Carmen Ugarte García, Asunción Yturbe Herranz

Coordinación Editorial
 Rafael Fernández Calvo <rfcalvo@ati.es>

Composición y autoedición
 Jorge Llácer

Administración
 Tomás Brunete, María José Fernández

SECCIONES TÉCNICAS: COORDINADORES

- Arquitecturas**
 Antonio Gonzalez Colás (DAC-UPC) <antonio@ac.upc.es>
- Bases de Datos**
 Coral Calero Muñoz, Mario G. Piattini Velthuis (Escuela Superior de Informática, UCLM) <Coral.Calero@uclm.es>, <mpiatini@inf-cr.uclm.es>
- Calidad del Software**
 Juan Carlos Granja (Universidad de Granada) <jcgranja@goliat.ugr.es>
- Derecho y Tecnologías**
 Isabel Hernando Collazos (Fac. Derecho de Donostia, UPV) <ihernando@legaltek.net>
- Enseñanza Universitaria de la Informática**
 Cristóbal Pareja Flores (Dep. Sistemas Informáticos y Programación-UCM) <cpareja@sip.ucm.es>
- Informática Gráfica**
 Roberto Vivó (Eurographics, sección española) <rvivo@dsic.upv.es>
- Ingeniería del Software**
 Luis Fernández (PRIS-E.L./UEM) <lufern@dpris.esi.uem.es>
- Inteligencia Artificial**
 Federico Barber, Vicente Botti (DSIC-UPV) <fvbotti, fbarber@dsic.upv.es>
- Interacción Persona-Computador**
 Julio Abascal González (FI-UPV) <julio@si.ehu.es>
- Internet**
 Alonso Álvarez García (TID) <alonso@ati.es>
 Llorenç Pagés Casas (Atlante) <pages@ati.es>
- Lengua e Informática**
 M. del Carmen Ugarte (IBM) <cugarte@ati.es>
- Lenguajes informáticos**
 Andrés Marín López (Univ. Carlos III) <amarin@it.uc3m.es>
 J. Ángel Velázquez (ESCET-URJC) <a.velazquez@escet.urjc.es>
- Libertades e Informática**
 Alfonso Escolán (PIR-Univ. de La Laguna) <aescolan@ull.es>
- Lingüística computacional**
 Xavier Gómez Guinovart (Univ. de Vigo) <jgomez@uvigo.es>
 Manuel Palomar (Univ. de Alicante) <mpalomar@dlsi.ua.es>
- Profesión informática**
 Rafael Fernández Calvo (ATI) <rfcalvo@ati.es>
 Miquel Sàrries Grinyó (Ayto. de Barcelona) <msarries@ati.es>
- Seguridad**
 Javier Areitio (Redes y Sistemas, Bilbao) <jareitio@orion.deusto.es>
- Sistemas de Tiempo Real**
 Alejandro Alonso, Juan Antonio de la Puente (DIT-UPM) <[alonso,jpuente]@dit.upm.es>
- Software libre**
 Jesús M. González Barahona, Pedro de las Heras Quirós (GSYC, URJC) <[jgb,pheras]@gsyc.escet.urjc.es>
- Tecnología de Objetos**
 Esperanza Marcos (URJC) <e.marcos@escet.urjc.es>
 Gustavo Rossi (LIFIA-UNLP, Argentina) <gustavo@sol.info.unpl.edu.ar>
- Tecnologías para la Educación**
 Benita Compostela (F. CC. PP. - UCM) <benita@diad.eunet.es>
 Josep Sales Rufí (ESPIRAL) <jsales@pie.xtec.es>
- Tecnologías y Empresa**
 Pablo Hernández Medrano <phmedrano@terra.es>
- TIC para la Sanidad**
 Valentín Masero Vargas (DI-UNEX) <vmasero@unex.es>

Las opiniones expresadas por los autores son responsabilidad exclusiva de los mismos. Novática permite la reproducción de todos los artículos, salvo los marcados con © o copyright, debiéndose en todo caso citar su procedencia y enviar a Novática un ejemplar de la publicación.

Coordinación Editorial y Redacción Central (ATI Madrid)
 Padilla 66, 3º, dcha., 28006 Madrid
 Tlf.914029391; fax.913093685 <novatica@ati.es>

Composición, Edición y Redacción ATI Valencia
 Palomino 14, 2º, 46003 Valencia
 Tlf./fax 963918531 <secreval@ati.es>

Administración, Suscripciones y Redacción ATI Cataluña
 Via Laietana 41, 1º, 1ª, 08003 Barcelona
 Tlf.934125235; fax 934127713 <secregen@ati.es>

Redacción ATI Andalucía
 Isaac Newton, s/n, Ed. Sadiel, Isla Cartuja 41092 Sevilla
 Tlf./fax 954460779 <secreand@ati.es>

Redacción ATI Aragón
 Lagasca 9, 3-B, 50006 Zaragoza
 Tlf./fax 976235181 <secreara@ati.es>

Redacción ATI Asturias-Cantabria <gp-astucant@ati.es>

Redacción ATI Castilla-La Mancha <gp-clmancha@ati.es>

Redacción ATI Galicia
 Recinto Ferial s/n, 36540 Silleda (Pontevedra)
 Tlf.986581413; fax 986580162 <secregal@ati.es>

Publicidad: Padilla 66, 3º, dcha., 28006 Madrid
 Tlf.914029391; fax.913093685 <novatica.publicidad@ati.es>

Imprenta: 9 Impressió S.A., Juan de Austria 66, 08005 Barcelona.
Depósito Legal: B 15.154-1975
ISBN: 0211-2124; CODEN NOVAEC

Portada: Antonio Crespo Foix / © ATI 2002

SUMARIO

En resumen: ¿eXtremismo? Sí, gracias **3**
Rafael Fernández Calvo

Monografía: «eXtreme Programming / Programación eXtrema»
 (En colaboración con **Informatik/Informatique** y **Upgrade**)

Editor invitado: *Luis Fernández Sanz*
Presentación: eXtreme Programming y la mejora en el desarrollo del software **5**

Luis Fernández Sanz
Referencias útiles sobre eXtreme Programming **7**

Luis Fernández Sanz
eXtreme Programming (XP): un nuevo método de desarrollo de software **8**

César F. Acebal, Juan M. Cueva Lovelle
La necesidad de velocidad: automatización de las pruebas de aceptación en un entorno de Programación Extrema **13**

Lisa Crispin, Tip House, con la contribución de *Carol Wade*
Estudios cualitativos sobre XP en una empresa de tamaño mediano **22**

Robert Gittins, Sian Hope, Ifor Williams
XP en proyectos complejos: algunas extensiones **27**

Martin Lippert, Stefan Roock, Henning Wolf, Heinz Züllighoven
XP e Ingeniería del Software: una opinión **32**

Luis Fernández Sanz
Programación Extremista **36**
Michael McCormick

Secciones Técnicas

Informática Gráfica
Modelado geométrico para visualización en tiempo real **39**
Inmaculada Remolar, José Ribelles, Óscar Belmonte, Miguel Chover, Cristina Rebollo

Interacción Persona-Computador
Modelos y herramientas para diseño y evaluación de la interfaz de usuario **44**
Fabio Paternò, Laila Paganelli, Carmen Santoro

Ingeniería Semiótica y comunicabilidad de las interfaces de usuario **49**
Clarisse Sieckenius de Souza

Software libre
Un entorno para enseñanza basado en software libre **55**
José Alfonso Accino

Tecnologías y Empresa
Knowledge Management en su organización **61**
Pablo Hernández Medrano

Referencias autorizadas **65**

Sociedad de la Información
Personal y transferible **70**

Mi opinión sobre las patentes de software en Europa
Jesús M. Gonzalez-Barahona

Programar es crear **71**
¿Queso!

25º Concurso Internacional de Programación ACM (2001): problema B **72**
«Configuración de un aeropuerto»: solución
Manuel Carro, Ángel Herranz, Julio Mariño, Pablo Sánchez

Asuntos Interiores

Programación de Novática **76**
Normas de publicación para autores / Socios Institucionales **77**

Monografía del próximo número: «Recuperación de la información y la Web»

Programar es crear

Traducción: Pablo Sánchez Torralba

<psanchez@skyrealms.org>

Érase una vez una polilla del queso llamada Amelia Chiche Masticadora que vivía en un enorme trozo de queso. Amelia debería estar tremendamente feliz puesto que estaba rodeada de delicioso queso, más del que nunca podría llegar a comerse. Sin embargo, sentía que en su vida faltaba algo.

Una mañana, cuando estaba soñando con queso, percibió un ruido que nunca antes había escuchado. Pero inmediatamente supo de qué se trataba —el sonido de una polilla macho, ¡royendo en el mismo trozo de queso!. (Averiguar el género de una polilla del queso sólo por el sonido de su ruido no es sencillo, pero todas las polillas del queso pueden hacerlo, debido a que sus padres también podían.)

Nada podía parar a Amelia ahora. Tenía que llegar hasta esa polilla lo antes posible, y para eso tenía que encontrar el camino más rápido. Amelia era capaz de roer un milímetro de queso en tan sólo diez segundos. Pero resulta que el camino recto no era también el más rápido. El queso en el que vivía Amelia estaba lleno de agujeros. Estos agujeros, que no eran más que burbujas de aire atrapadas en el queso, solían tener forma esférica. Pero en ocasiones, esos agujeros esféricos se solapaban, creando agujeros de muy diversas formas. Amelia era capaz de atravesar uno de esos agujeros en un tiempo prácticamente cero, ya que podía volar de un extremo al otro instantáneamente. Por ello podría ser recomendable atravesar agujeros para llegar hasta la otra polilla rápidamente.

Para este problema debes escribir un programa que, dadas las posiciones de ambas polillas y las de los agujeros en el queso, calcule el tiempo mínimo que necesitaría Amelia para llegar hasta la otra polilla. Para este problema se debe suponer que el queso es infinitamente grande. Por tanto Amelia no necesita abandonar el queso para llegar hasta la otra polilla (en especial porque los depredadores de polillas del queso la podrían devorar). También se supone que la otra polilla ha percibido la llegada de Amelia y no se mueve mientras ella está en camino.

Entrada

El fichero de entrada contiene la descripción de varios casos. Cada caso comienza con una línea que contiene un único entero n ($0 \leq n \leq 100$), que es el número de agujeros en el queso. Le siguen n líneas con cuatro enteros $x_i, y_i, z_i,$

¡Queso!

25º Concurso Internacional de Programación ACM (2001): problema B

r_i cada una. Estos describen el centro (x_i, y_i, z_i) y el radio r_i ($r_i > 0$) de los agujeros. Todos estos valores, y los siguientes, vienen dados en milímetros.

La descripción concluye con dos líneas con tres enteros cada una. La primera contiene los valores x_A, y_A, z_A , que indica la posición de Amelia en el queso. La segunda línea contiene x_0, y_0, z_0 , la posición de la otra polilla.

El fichero de entrada termina con una línea con el número -1.

Salida

Para cada caso de entrada, imprime una línea de salida, con el formato del ejemplo. Primero imprime el número del caso, comenzando en 1. A continuación el tiempo mínimo, en segundos, que necesita Amelia para llegar hasta la otra polilla, redondeado al entero más cercano. La entrada será tal que el redondeo no sea ambiguo.

Ejemplo de entrada Salida para el ejemplo de entrada

1	Cheese 1: Travel time = 100 sec
20 20 20 1	Cheese 2: Travel time = 20 sec
0 0 0	
0 0 10	
1	
5 0 0 4	
0 0 0	
10 0 0	
-1	

Nota del traductor: «*Say cheese*», alude a la manera de solicitar una sonrisa para una foto. Por tanto, en castellano debería haber sido «¡Patata!», pero Amelia, la protagonista de la historia, es un insecto del queso, y por ello se ha respetado el título original.