

**Novática**, revista fundada en 1975 y decana de la prensa informática española, es el órgano oficial de expresión y formación continua de **ATI** (Asociación de Técnicos de Informática), organización que edita también la revista **REICIS** (Revista Española de Innovación, Calidad e Ingeniería del Software).

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

ATI es miembro fundador de **CEPIS** (Council of European Professional Informatics Societies) y es representante de España en **IFIP** (International Federation for Information Processing); tiene un acuerdo de colaboración con **ACM** (Association for Computing Machinery), así como acuerdos de vinculación o colaboración con **AdaSpain**, **AIZ**, **ASTIC**, **RITSI** e **Hispalinux**, junto a la que participa en **Prolinova**.

#### Consejo Editorial

Guillem Alsina González, Rafael Fernández Calvo (presidente del Consejo), Jaime Fernández Martínez, Luis Fernández Sanz, José Antonio Gutiérrez de Mesa, Silvia Leal Martín, Didac López Vilas, Francesc Noguera Puig, Joan Antoni Pastor Collado, Andrés Pérez Payeras, Viktu Pons i Colomer, Moisés Robles Gener, Cristina Vigil Díaz, Juan Carlos Vigo López

#### Coordinación Editorial

Llorenç Pagés Casas < [lpages@ati.es](mailto:lpages@ati.es) >

#### Composición y autedición

Jorge Lácer Gil de Ranales

#### Traducciones

Grupo de Lengua e Informática de ATI < <http://www.ati.es/gt/lengua-informatica/> >

#### Administración

Tomás Brunete, María José Fernández, Enric Camarero

#### Secciones Técnicas - Coordinadores

##### Acceso y recuperación de la información

José María Gómez Hidalgo (Optenet), < [jmgomez@yaho.com](mailto:jmgomez@yaho.com) >

Manuel J. María López (Universidad de Huelva), < [manuel.maria@diecia.uhu.es](mailto:manuel.maria@diecia.uhu.es) >

##### Administración Pública electrónica

Francisco López Crespo (MAE), < [flc@ati.es](mailto:flc@ati.es) >

Sebastià Justicia Pérez (Diputación de Barcelona) < [sjusticia@ati.es](mailto:sjusticia@ati.es) >

##### Arquitecturas

Enrique F. Torres Moreno (Universidad de Zaragoza), < [enrique.torres@unizar.es](mailto:enrique.torres@unizar.es) >

José Filich Cardo (Universidad Politécnica de Valencia), < [jfilich@disca.upv.es](mailto:jfilich@disca.upv.es) >

##### Auditoría SITIC

Marina Tourinho Troilillo, < [marinatourinho@marinatourinho.com](mailto:marinatourinho@marinatourinho.com) >

Sergio Gómez-Landero Pérez (Endesa), < [sergio.gomezlandero@endesa.es](mailto:sergio.gomezlandero@endesa.es) >

##### Derecho y tecnologías

Isabel Hernando Collazos (Fac. Derecho de Donostia, UPV), < [isabel.hernando@ehu.es](mailto:isabel.hernando@ehu.es) >

Elena Davara Fernández de Marcos (Davara & Davara), < [edavara@davara.com](mailto:edavara@davara.com) >

##### Enseñanza Universitaria de la Informática

Cristóbal Parsja Flores (DSIP-UCM), < [cparsja@sip.ucm.es](mailto:cparsja@sip.ucm.es) >

J. Ángel Velázquez Iturbide (DLSI1, URJC), < [angel.velazquez@urjc.es](mailto:angel.velazquez@urjc.es) >

##### Entorno digital personal

Andrés Marín López (Univ. Carlos III), < [amarin@it.uc3m.es](mailto:amarin@it.uc3m.es) >

Diego Gachet Pérez (Universidad Europea de Madrid), < [gachet@uem.es](mailto:gachet@uem.es) >

##### Estándares Web

Encarna Quesada Ruiz (Virati), < [encarna.quesada@virati.com](mailto:encarna.quesada@virati.com) >

José Carlos del Arco Prieto (TOP Sistemas e Ingeniería), < [jcarco@gmail.com](mailto:jcarco@gmail.com) >

##### Gestión del Conocimiento

Juan Baiget Solé (Cap Gemini Ernst & Young), < [joan.baiget@ati.es](mailto:joan.baiget@ati.es) >

##### Gobierno corporativo de las TI

Manuel Palao García-Suñeto (ATI), < [manuel@palao.com](mailto:manuel@palao.com) >

Miguel García-Monreal (ITI) < [mgarciamonreal@ititrends.institute.org](mailto:mgarciamonreal@ititrends.institute.org) >

##### Informática y Filosofía

José Ángel Olivares Varela (Escuela Superior de Informática, UCLM), < [joseangel.olivares@uclm.es](mailto:joseangel.olivares@uclm.es) >

Roberto Feltrero Dreja (UNED), < [rfeltrero@gmail.com](mailto:rfeltrero@gmail.com) >

##### Informática Gráfica

Miguel Chover Sellés (Universitat Jaume I de Castellón), < [mchover@lsi.uji.es](mailto:mchover@lsi.uji.es) >

Roberto Vivó Hernández (Eurographics, sección española), < [rvivo@dsic.upv.es](mailto:rvivo@dsic.upv.es) >

##### Ingeniería del Software

Javier Dotado Cosin (DLSI-UPV), < [ddotado@si.uhu.es](mailto:ddotado@si.uhu.es) >

Daniel Rodríguez García (Universidad de Alcalá), < [daniel.rodriguez@uah.es](mailto:daniel.rodriguez@uah.es) >

##### Inteligencia Artificial

Vicente Boti Navarro, Vicente Julián Inglada (DSIC-UPV), < [vbotti.vinglada@dsic.upv.es](mailto:vbotti.vinglada@dsic.upv.es) >

##### Interacción Persona-Computador

Pedro M. Latorre Andrés (Universidad de Zaragoza, AIPO), < [platorre@unizar.es](mailto:platorre@unizar.es) >

Francisco L. Gutiérrez Vela (Universidad de Granada, AIPO), < [fgutierrez@ugr.es](mailto:fgutierrez@ugr.es) >

##### Lengua e Informática

M. del Carmen Ugarte García (ATI), < [cugarte@ati.es](mailto:cugarte@ati.es) >

##### Lenguajes Informáticos

Oscar Belmonte Fernández (Univ. Jaime I de Castellón), < [obelfern@lsi.uji.es](mailto:obelfern@lsi.uji.es) >

Inmaculada Coma Tany (Univ. de Valencia), < [inmaculada.coma@uv.es](mailto:inmaculada.coma@uv.es) >

##### Lingüística computacional

Xavier Gómez Guinovart (Univ. de Vigo), < [xg@uvigo.es](mailto:xg@uvigo.es) >

Manuel Palomar (Univ. de Alicante), < [mpalomar@disi.ua.es](mailto:mpalomar@disi.ua.es) >

##### Mundo estudiantil y jóvenes profesionales

Federico G. Mon Trotti (ATI), < [fgmon@ati.es](mailto:fgmon@ati.es) >

Mikel Salazar Peña (Área de Jóvenes Profesionales, Junta de ATI Madrid), < [mikelbo\\_uni@yahoo.es](mailto:mikelbo_uni@yahoo.es) >

##### Profesión Informática

Rafael Fernández Calvo (ATI), < [rfcalvo@ati.es](mailto:rfcalvo@ati.es) >

Miguel Sarrías Grifó (ATI), < [miguel@sarrias.net](mailto:miguel@sarrias.net) >

##### Redes y servicios telemáticos

José Luis Marzo Lázaro (Univ. de Girona), < [joseluis.marzo@udg.es](mailto:joseluis.marzo@udg.es) >

Juan Carlos López López (UCLM), < [juancarlos.lopez@uclm.es](mailto:juancarlos.lopez@uclm.es) >

##### Robótica

José Cortés Arenas (Sopra Group), < [joscortea@gmail.com](mailto:joscortea@gmail.com) >

Juan González Gómez (Universidad CARLOS III), < [juan@iearobotics.com](mailto:juan@iearobotics.com) >

##### Seguridad

Javier Arellano Bertolin (Univ. de Deusto), < [jarellito@deusto.es](mailto:jarellito@deusto.es) >

Javier López Muñoz (ETSI Informática-UMA), < [jlm@lcc.uma.es](mailto:jlm@lcc.uma.es) >

##### Sistemas de Tiempo Real

Alejandro Alonso Muñoz, Juan Antonio de la Puente Alfaro (DIT-UPM), < [falonso.puente@dit.upm.es](mailto:falonso.puente@dit.upm.es) >

##### Software Libre

Jesus M. González Barahona (GSYC - URJC), < [jgb@gsyc.es](mailto:jgb@gsyc.es) >

Israel Hénarez Tabernero (Universidad Politécnica de Madrid), < [isra@herraz.org](mailto:isra@herraz.org) >

##### Tecnología de Objetos

Jesus Garcia Molina (DIS-UM), < [jmolina@um.es](mailto:jmolina@um.es) >

Gustavo Rossi (LIFIA-UNLP Argentina), < [gustavo@sol.inf.unlp.edu.ar](mailto:gustavo@sol.inf.unlp.edu.ar) >

##### Tecnologías para la Educación

Juan Manuel Dodero Berardo (UC3M), < [dodero@inf.uc3m.es](mailto:dodero@inf.uc3m.es) >

César Pablo Córcoles Briongo (UOC), < [ccorcoles@uoc.edu](mailto:ccorcoles@uoc.edu) >

##### Tecnologías y Empresa

Didac López Vilas (Universitat de Girona), < [didac.lopez@ati.es](mailto:didac.lopez@ati.es) >

Alonso Álvarez García (TID), < [agag@tid.es](mailto:agag@tid.es) >

##### Tendencias tecnológicas

Gabriel Martí Fuentes (Interbits), < [gabi@atinet.es](mailto:gabi@atinet.es) >

Juan Carlos Vigo (ATI) < [juancarlosvigo@atinet.es](mailto:juancarlosvigo@atinet.es) >

##### TIC y Turismo

Andrés Aguayo Maldonado, Antonio Guevara Plaza (Univ. de Málaga), < [aguayo.guevara@lcc.uma.es](mailto:aguayo.guevara@lcc.uma.es) >

Las opiniones expresadas por los autores son responsabilidad exclusiva de los mismos. **Novática** permite la reproducción, sin ánimo de lucro, de todos los artículos, a menos que lo impida la modalidad de © o copyright elegida por el autor, debiéndose en todo caso citar su procedencia y enviar a **Novática** un ejemplar de la publicación.

#### Coordinación Editorial, Redacción Central y Redacción ATI Madrid

Plaza de España 6, 2ª planta, 28008 Madrid

Tfno. 91 4029391; fax: 91 3093685 < [novatica@ati.es](mailto:novatica@ati.es) >

#### Composición, Edición y Redacción ATI Valencia

Av. del Reino de Valencia 23, 46005 Valencia

Tfno. 96 374 0173 < [novatica\\_valencia@ati.es](mailto:novatica_valencia@ati.es) >

#### Administración y Redacción ATI Cataluña

Calle Avila 50, 3a planta, local 9, 08005 Barcelona

Tfno. 93 4125235; fax: 93 4127173 < [secretgen@ati.es](mailto:secretgen@ati.es) >

#### Redacción ATI Andalucía

< [secretand@ati.es](mailto:secretand@ati.es) >

#### Redacción ATI Galicia

< [secretgal@ati.es](mailto:secretgal@ati.es) >

#### Subscripción y Ventas

< [novatica.subscripciones@atinet.es](mailto:novatica.subscripciones@atinet.es) >

#### Publicidad

Plaza de España 6, 2ª planta, 28008 Madrid

Tfno. 91 4029391; fax: 91 3093685 < [novatica@ati.es](mailto:novatica@ati.es) >

Imprenta: Derra S.A., Juan de Austria 66, 08005 Barcelona.

Depósito legal: B. 15.154-1975 - ISSN: 0211-2124; CODEN NOVAEC

Portada: "Contorno natural" - Concha Ariza Pérez / © ATI

Diseño: Fernando Agrestia / © ATI 2003

## editorial

**En el proceso de cambios**

> 02

**en resumen**

**Fin de año en clave optimista**

> 02

*Llorenç Pagés Casas*

**noticias de CLEI**

**Resumen del Congreso y de la Asamblea de CLEI 2013**

> 03

*Ramon Puigjaner Trepal*

**noticias de CEPIS**

**Francisco López Crespo galardonado con el ICT Professionalism Award**

> 03

**noticias de IFIP**

**Reunión del TC1 (Foundations of Computer Science)**

> 04

*Jacques Sakarovitch, Joaquim Gabarró Vallés*

**actividades de ATI**

**Entrega del Premio Novática 2012**

> 04

## monografía

**Empresa 2.0: Últimos resultados de I+D**

Editores invitados: *Joaquín Peña Siles, Manuel Resinas Arias de Reyna*

**Presentación. Empresa 2.0: Últimos resultados de I+D en RRHH, Marketing, Comunicación y Software**

> 05

*Joaquín Peña Siles, Manuel Resinas Arias de Reyna*

**Estrategia de marketing para el ingreso de la pequeña empresa**

**en las redes sociales**

> 08

*Joaquín Martín-Albo Mateos-Aparicio, Coral Calero Muñoz*

**Incidencia de las TIC como estrategia de comunicación en organizaciones culturales**

> 15

*Luz Madelén Ramírez Guzmán, Luisa Fernanda Henao Castaño*

**Percepción de valor de marca online en compañías aéreas de low cost.**

**El caso de Vueling, Easyjet y Ryanair**

> 18

*Ana María López Fernández, Ángela Escobar Ubrega, Ángel Francisco Villarejo Ramos*

**Los códigos de conducta como elemento facilitador en la difusión de la Web 2.0**

> 23

*Salvador Bueno Avila, María Dolores Gallego Pereira, David López Jiménez*

**Un estudio de la confianza inicial hacia el líder de equipos de trabajo virtuales**

> 29

*Miguel Guinaliu Blasco, Pau Jordan Blasco*

**Confianza y desconfianza en redes sociales: Detección de trolls**

> 34

*F. Javier Ortega, Fermín L. Cruz, José A. Troyano, Carlos G. Vallejo, Fernando Enriquez*

## secciones técnicas

**Profesión informática**

**Participación de la mujer en Computación: su presencia e influencia en la Universidad Simón Bolívar**

> 40

*Rosseline Rodríguez, Soraya Carrasquel*

**Redes y servicios telemáticos**

**VQoS: Una herramienta visual para la evaluación de protocolos de enrutamiento ad hoc**

> 48

*Amaia Aguirregoitia, Begoña Blanco Jauregui*

**Tecnologías y Empresa**

**Automatizando la gestión del riesgo tecnológico**

> 56

*Santiago Moral Rubio, Juan Manuel Vara Mesa, Marcos López Sanz,*

*Rosa María Quintanar Leal, Esperanza Marcos Martínez*

**TIC y Turismo**

**Sistemas de recomendación basados en grupo para su aplicación en realidad aumentada**

> 63

*José L. Leiva Olivencia, Antonio Guevara Plaza, Carlos Rossi Jiménez,*

*Andrés Aguayo Maldonado*

**Referencias autorizadas**

> 68

**sociedad de la información**

**Distinciones profesionales**

**Laudatio a Mateo Valero Cortés**

> 75

*Francisco Tirado Fernández*

**Discurso de investidura**

> 77

*Mateo Valero Cortés*

**Programar es crear**

**El problema de las tablas**

> 80

**(Competencia UTN-FRC 2012, enunciado)**

*Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas*

**El problema de la carrera de autos**

> 81

**(Competencia UTN-FRC 2012, solución)**

*Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas*

## Asuntos Interiores

**Coordinación editorial / Programación de Novática / Socios Institucionales**

> 83

Tema del próximo número:

**"Eficiencia energética en centros de procesos de datos"**

Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas

Laboratorio de Investigación de Software MsLabs, Dpto. Ing. en Sistemas de Información, Facultad Regional Córdoba - Universidad Tecnológica Nacional (Argentina)

<jotacastillo@gmail.com>,  
<diegojserrano@gmail.com>,  
<ing.marinacardenas@gmail.com>

En el problema de la carrera de autos, cada auto presenta una velocidad constante, por lo cual se conoce con antelación cuál será el ganador. El ranking de los demás competidores se determina cuando el ganador finaliza su última vuelta en base a las posiciones en las que se encuentran en el circuito, tomando el ganador como referencia. Entonces, el segundo puesto es aquel que se encuentra más cerca del ganador, y así sucesivamente.

Este problema fue resuelto utilizando el paradigma orientado a objetos (POO), lo cual nos permite codificar una solución corta, y fácil de comprender.

En la solución se pueden observar 4 clases: `CarreraAutos`, `Corredor`, `ComparadorVelocidad`, y `ComparadorDistancia`. La primera clase es la que contiene el método estático `main` y es el que se ejecutará para iniciar el proceso. A su vez, esta clase presenta el método `iniciar()` que se encarga de leer la entrada del programa y de crear los corredores que participan de la carrera de autos.

La clase `Corredor` es la que contiene toda la información relativa a los corredores y que, siguiendo el POO, se convertirán en atributos de la clase. Esta información está formada por el número de corredor, distancia al piloto ganador, velocidad en metros por segundo, cantidad de vueltas realizadas en la carrera, y el nombre del piloto. De estos atributos, la distancia y la cantidad de vueltas son atributos calculados, mientras que los restantes son datos de entrada.

En esta clase se distingue el método `calcularDistancia()` que le permite conocer a cada corredor su distancia respecto al corredor que finalizó primero en la carrera. Por otra parte, la clase `ComparadorVelocidad` implementa un comparador de corredores, y será utilizada para conocer el corredor ganador.

Se puede observar que la solución aquí propuesta hace uso de las facilidades y herramientas algorítmicas que brinda el lenguaje para poder resolver el problema planteado. Así, se utilizan colecciones, y en particular `ArrayList`, como estructuras de datos, utilizando los métodos `min` y `sort` de `Collections` que son algoritmos previa-

## El problema de la carrera de autos

El enunciado de este problema apareció en el número 224 de *Novática* (julio-agosto 2013, p.77).

mente implementados. Entonces, uno de los beneficios de trabajar con un `ArrayList` es utilizar métodos de búsqueda y ordenamiento, por lo que se utilizó el método `Collections.min` que permite comparar las velocidades de los corredores (a través de un `Comparator`), lo cual es codificado por la clase `ComparadorVelocidad`.

En cuanto a su complejidad, notamos que `min` presenta un orden lineal y `sort` presenta  $n \cdot \log n$  en caso promedio empleando una variante del algoritmo *Mergesort*. Asimismo, una posible mejora consistiría en aprovechar el procedimiento de lectura de la entrada para tomar el máximo de las velocidades y así conocer el ganador.

Cabe aclarar que la mayoría de los lenguajes de programación presentan estructuras de datos equivalentes a los `ArrayList` de Java, y acceso a algoritmos de búsqueda y/o de ordenamiento implementados como parte de la biblioteca de clases de cada lenguaje. Igualmente, en caso de no contar con las mismas, no debería presentar dificultad codificar algún algoritmo de ordenamiento para poder resolver este problema.

Siguiendo la secuencia de ejecución del programa propuesto, lo primero que se realiza es la lectura de los datos de entrada, se crea cada uno de los `Corredores` y se los agrega al `ArrayList` de `Corredores`. A continuación, se obtiene el corredor ganador en base a la diferencia de velocidades con los demás corredores.

Luego, se realiza el recorrido del `ArrayList` para asignarle a cada corredor (distinto del ganador) su distancia respecto al corredor ganador. Esto es llevado a cabo por el método `calcularDistancia()` de la clase `Corredor`.

Finalmente, ya que el enunciado solicita mostrar todos los corredores ordenados por distancia, se hace uso del método `Collections.sort`, el cual también necesita un `Comparator` y que es provisto por la clase `ComparadorDistancia`.

Para lograr la salida como la requiere el enunciado del problema, se ha codificado el método `toString()` de la clase `Corredor`, de manera tal que un recorrido simple permita

visualizar toda la salida correctamente formateada.

A continuación se expone el código de la solución del problema en el lenguaje de programación Java.

```
import java.util.*;

public class CarreraAutos {

    public static void main(String[] args) {
        new CarreraAutos().iniciar();
    }

    public void iniciar() {
        Scanner sc = new Scanner(System.in);

        int C, P, L, T;
        C = sc.nextInt();

        ArrayList<Corredor> corredores = new ArrayList<>();
        for (int i = 0; i < C; i++) {
            P = sc.nextInt();
            L = sc.nextInt();
            T = sc.nextInt();

            corredores.clear();
            for (int j = 0; j < P; j++) {
                Corredor c = new Corredor();
                c.numero = sc.nextInt();
                c.nombre = sc.next();
                c.velocidad = sc.nextInt();
                corredores.add(c);
            }

            Corredor ganador = Collections.min(corredores, new ComparadorVelocidad());
            ganador.distancia = 0;
            ganador.vueltas = T;
            for (Corredor c : corredores) {
                if (c != ganador) {
                    c.calcularDistancia(L, T, ganador);
                }
            }

            Collections.sort(corredores, new ComparadorDistancia());
            for (Corredor c : corredores) {
                System.out.println(c);
            }
            System.out.println();
        }
    }

    class Corredor {
        int numero, distancia, velocidad, vueltas;
        String nombre;

        @Override
        public String toString() {
            return nombre + " " + distancia + " " + vueltas;
        }

        /**
         * Este es el método central de la solución. Calcula la distancia
         * entre el corredor y el final del circuito cuando el ganador
         * finaliza la última vuelta
         * @param longitud Es la longitud del circuito
         * @param vueltasTotales Es la cantidad de vueltas previstas
         * @param ganador Es la referencia al ganador de la carrera
         */
        public void calcularDistancia(int longitud, int vueltasTotales, Corredor ganador) {
            double tiempoGanador = (double) (longitud * vueltasTotales) / ganador.velocidad;
            int distanciaTotal = (int) (tiempoGanador * this.velocidad);
            this.vueltas = distanciaTotal / longitud;
            this.distancia = longitud - (distanciaTotal % longitud);
        }
    }

    class ComparadorVelocidad implements Comparator<Corredor> {
        @Override
        public int compare(Corredor o1, Corredor o2) {
            return o2.velocidad - o1.velocidad;
        }
    }

    class ComparadorDistancia implements Comparator<Corredor> {
        @Override
        public int compare(Corredor o1, Corredor o2) {
            return o1.distancia - o2.distancia;
        }
    }
}
```