

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**, **AI2**, **ASTIC**, **RITSI** e **Hispalinux**, junto a la que participa en **Prolnova**.

Consejo Editorial

Ignacio Aguiló Sousa, Guillem Alsina González, María José Escalona Cuaremas, Rafael Fernández Calvo (presidente del Consejo), Jaime Fernández Martínez, Luis Fernández Sanz, Didac López Viñas, Celestino Martín Alonso, José Onofre Montes Andrés, Francesc Noguera Puig, Ignacio Pérez Martínez, Andrés Pérez Payeras, Víkto Pons i Colomer, Juan Carlos Vigo López

Coordinación Editorial

Llorenç Pagés Casas <pagues@ati.es>

Composición y autoedición

Jorge Llácer Gil de Ramales

Traducciones

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

Administración

Tomás Brunete, María José Fernández, Enric Camarero, Felicidad López

Secciones Técnicas - Coordinadores

Acceso y recuperación de la información

José María Gómez Hidalgo (Optenet), <jmgomez@yahoos.es>

Manuel J. María López (Universidad de Huelva), <manuel.maria@diehsia.uhu.es>

Administración Pública electrónica

Francisco López Crespo (MAE), <flc@ati.es>

Sebastià Justicia Pérez (Diputación de Barcelona), <sjusticia@ati.es>

Arquitecturas

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

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

Auditoría SITIC

Marina Tourino Trolitio, <marinatourino@marinatourino.com>

Manuel Palao García-Suñito (ATI), <manuel@palao.com>

Derecho y tecnologías

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

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

Enseñanza Universitaria de la Informática

Cristóbal Pareja Flores (DSIP-UCM), <cpareja@sip.ucm.es>

J. Ángel Velázquez Iturbide (DLSI, URJC), <angel.velazquez@urjc.es>

Entorno digital personal

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

Diego Gachet Páez (Universidad Europea de Madrid), <gachet@uem.es>

Estandares Web

Encarna Quesada Ruiz (Virati), <encarna.quesada@virati.com>

José Carlos del Arco Prieto (TCP Sistemas e Ingeniería), <jcarco@gmail.com>

Gestión del Conocimiento

Juan Baiget Solé (Cap Gemini Ernst & Young), <juan.baiget@ati.es>

Informática y Filosofía

José Ángel Olivás Varela (Escuela Superior de Informática, UCLM), <josangel.olivas@uclm.es>

Roberto Feltrero Oreja (UNED), <rfeltrero@gmail.com>

Informática Gráfica

Miguel Chover Selles (Universitat Jaume I de Castellón), <mchover@lsi.uji.es>

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

Ingeniería del Software

Javier Dolado Cosín (DLSI-UPV), <dolado@si.ehu.es>

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

Inteligencia Artificial

Vicente Boti Navarro, Vicente Julián Inglada (DSIC-UPV), <vbotti@vinglada.com>

Interacción Persona-Computador

Pedro M. Latore Andrés (Universidad de Zaragoza, AIPO), <platore@unizar.es>

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

Lengua e Informática

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

Lenguajes Informáticos

Oscar Belmonte Fernández (Univ. Jaime I de Castellón), <belferm@lsi.uji.es>

Inmaculada Coma Tatay (Univ. de Valencia), <inmaculada.coma@uv.es>

Lingüística computacional

Xavier Gómez Guinovart (Univ. de Vigo), <xgg@uvigo.es>

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

Mundo estudiantil y jóvenes profesionales

Federico G. Mon Trotti (RITSI), <gmon.trotti@gmail.com>

Mikel Salazar Peña (Asociación Jóvenes Profesionales, Junta de ATI Madrid), <mikelbto_uni@yahoo.es>

Profesión Informática

Rafael Fernández Calvo (ATI), <rfoalvo@ati.es>

Miguel Sarrías Grilo (ATI), <mqsarrias@ati.es>

Redes y servicios telemáticos

José Luis Marco Lázaro (Univ. de Girona), <joseluis.marco@udg.es>

Juan Carlos López López (UCLM), <juancarlos.lopez@uclm.es>

Robótica

José Cortés Arenas (Sopra Group), <jccortes@gmail.com>

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

Seguridad

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

Javier López Muñoz (ETS Informática-UMA), <jlm@icc.uma.es>

Sistemas de Tiempo Real

Alejandro Alonso Muñoz, Juan Antonio de la Fuente Alfaro (DIT-UPM), <gaalonso@puente@dit.upm.es>

Software Libre

Jesús M. González Barahona (GSYC - URJC), <jgb@gsyc.es>

Israel Herráiz Taberner (Universidad Politécnica de Madrid), <isra@herraz.org>

Tecnología de Objetos

Jesús García Moine (DIS-UI), <jmolina@um.es>

Gustavo Rossi (LIFIA-UNLP Argentina), <gustavo@sol.info.unlp.edu.ar>

Tecnologías para la Educación

Juan Manuel Dodero Beardo (UC3M), <dodero@inf.uc3m.es>

César Pablo Córcoles Briongo (UOC), <ccorcoles@uoc.edu>

Tecnologías y Empresa

Didac López Viñas (Universitat de Girona), <didac.lopez@ati.es>

Francisco Javier Cantais Sánchez (Indra Sistemas), <fcantais@gmail.com>

Tendencias tecnológicas

Alonso Álvarez García (TID), <aad@tid.es>

Gabriel Martí Fuentes (Interbits), <gabi@atinet.es>

TIC y Turismo

Andrés Aguayo Maldonado, Antonio Guevara Plaza (Univ. de Málaga), <aguayo.guevara@icc.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

Padilla 66, 3º dcha., 28006 Madrid

Tfno. 914029391; fax 913093685 <novatica@ati.es>

Composición, Edición y Redacción ATI Valencia

Av. del Reino de Valencia 23, 46005 Valencia

Tfno. 963740173 <novatica_val@ati.es>

Administración y Redacción ATI Cataluña

Via Laietana 46, ppal. 1º, 08003 Barcelona

Tfno. 934125235; fax 934127713 <secretgen@ati.es>

Redacción ATI Aragón

Lagasca 3, 5º B., 50006 Zaragoza

Tfno. fax 916238181 <secretara@ati.es>

Redacción ATI Andalucía

<secretand@ati.es>

Redacción ATI Galicia

<secretgal@ati.es>

Subscripción y Ventas

<novatica.subscriptions@atinet.es>

Publicidad

Padilla 66, 3º dcha., 28006 Madrid

Tfno. 914029391; fax 913093685 <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: Lenguaje primario - Concha Arias Pérez / © ATI

Diseño: Fernando Agresta / © ATI 2003

Nº 222, marzo-abril 2013, año XXXIX

editorial

Una iniciativa de creación de empleo para los profesionales TIC > 02

en resumen

Estudiantes antiguos y jóvenes profesionales > 02

Llorenç Pagés Casas

noticias de IFIP

TC2: Grupos de trabajo y llamamiento a la participación > 03

Antonio Vallecillo Moreno

monografía

Lenguajes de programación

Editores invitados: *Óscar Belmonte Fernández* y *Carlos Granell Canut*

Presentación. Lenguajes de programación en perspectiva > 04

Óscar Belmonte Fernández, Carlos Granell Canut

Los lenguajes de programación en perspectiva

> 09

Ricardo Peña Mari

La programación funcional

> 14

Manuel Montenegro Montes

Estándares en la web

> 20

Carlos Blé Jurado

Laudatio a Antony R. Hoare

> 24

Ricardo Peña Mari

Respuesta a la Laudatio

> 26

Antony R. Hoare

secciones técnicas

Enseñanza Universitaria de la Informática

Vídeo-ejercicios didácticos para el aprendizaje de la programación > 28

Germán Moltó

Seguridad

Análisis de Bitcoin: Sistema P2P de pago digital descentralizado con moneda > 34

Javier Arellano Bertolin

Software Libre

Monitorización de PostgreSQL: Plugin para Pandora FMS > 42

Luis Caballero Cruz

Tecnologías para la Educación

Animaciones adaptativas de programas: una propuesta basada en estilos > 49

Francisco Manso-González, Jaime Urquiza Fuentes, Estefanía Martín Barroso, Marta Gómez-Gómez

TIC y Turismo

Extracción automática de fichas de recursos turísticos de la web > 55

Iker Manterola Isasa, Xabier Saralegi Urizar, Sonia Bilbao Arechabala

Referencias autorizadas

> 60

Sociedad de la Informática

Privacidad y nuevas tecnologías

Privacidad y vigilancia: Una guía básica > 67

Aaron Martín

Programar es crear

El problema del CUIT > 74

(Competencia UTN-FRC 2012, problema D, enunciado)

Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas

El problema del Buscaminas Cuadrado en 3D

(Competencia UTN-FRC 2012, problema F, solución) > 75

Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas

Asuntos Interiores

Coordinación editorial / Programación de Novática / Socios Institucionales > 77

Tema del próximo número: **"Minería de procesos"**

Ricardo Peña Marí

Catedrático del departamento de Sistemas Informáticos y Computación, Universidad Complutense de Madrid

<ricardo@sip.ucm.es>

Laudatio a Antony R. Hoare

Excelentísimo Sr. Rector Magnífico, Excelentísimas e Ilustrísimas Autoridades Académicas, Señores Profesores, Personal de Administración y Servicios, Estudiantes, Señoras y Señores

Sir Charles Antony Richard Hoare, conocido comúnmente entre los informáticos como Tony Hoare, es uno de los científicos que más huella han dejado en nuestra disciplina. En su larga vida científica, ha sentado los fundamentos de muchas de las teorías y técnicas que hoy enseñamos en la facultades de Informática de todo el mundo. Sus aportaciones han contribuido decisivamente a hacer de la programación de computadores una disciplina científicamente fundada, partiendo de una situación inicial en la que se parecía más a un arte, a un oficio artesanal, o en sus propias palabras, a una brujería, que a una ciencia.

Participó en los años sesenta en el diseño de los primeros lenguajes de programación estructurados, sentó las bases de la verificación formal de programas, estableció mecanismos de programación para la construcción fiable de sistemas operativos, creó modelos matemáticos para el razonamiento sobre programas concurrentes, y en los últimos años está dedicado a establecer relaciones entre distintos formalismos, tratando de crear una teoría unificadora de la programación.

Para los estudiantes de informática, Tony Hoare es el creador del famoso *Quicksort*, un algoritmo de ordenación cuya eficiencia promedio supera a la de todos los otros algoritmos de ordenación previos y posteriores.

Este Doctorado Honoris Causa pretende humildemente reconocer esta dilatada y fructífera trayectoria.

Tony Hoare nació en Colombo, capital de la antigua Ceilán, y se graduó en Lenguas Clásicas y Filosofía por la Universidad de Oxford. Comenzó trabajando en la Universidad Estatal de Moscú, donde ideó el algoritmo *Quicksort*. Entre 1960 y 1968 trabajó para la empresa británica Elliot Computers, una de las primeras compañías europeas que fabricaban computadores científicos. Allí dirigió el desarrollo de un compilador para el lenguaje Algol-60, y se involucró en el diseño de sus lenguajes sucesores: Algol-W, y Simula-67. En 1968 pasó a la universidad y fue profesor de la *Queen University*

El pasado día 10 de mayo de 2013, en el Salón de Actos de la Facultad de Informática, Antony R. Hoare fue investido Doctor *Honoris Causa* por la Universidad Complutense de Madrid. Publicamos a continuación el discurso de *laudatio* que pronunció Ricardo Peña Marí durante la ceremonia.

of Belfast. A partir de 1977, pasó a liderar el *Programming Research Group* de la Universidad de Oxford, donde transcurrió la mayor parte de su carrera profesional. Actualmente es Investigador Principal en el *Microsoft Research Center* de Cambridge.

Su artículo de 1969 "*An axiomatic basis for computer programming*", visto en perspectiva, puede considerarse como el equivalente a las Leyes de Newton de la programación. Por primera vez se enunciaban unas leyes lógicas que permitían comprender el significado de los programas independientemente de las máquinas que los ejecutan. Su lógica, llamada en su honor *Lógica de Hoare*, sentó las bases para el razonamiento matemático sobre los programas y continúa siendo usada en la actualidad, tanto manualmente como mediante el uso de herramientas. Con ella se ha descrito la semántica de numerosos lenguajes de programación.

Estos años, los finales de los 60, son cruciales para la historia de la programación. Son años de fracasos estrepitosos en el desarrollo de grandes sistemas software, como el del paradigmático sistema operativo OS/360 de IBM. Se habla abiertamente de una "crisis del software", y el nombre "Ingeniería del Software" dado a una famosa conferencia patrocinada por la OTAN en 1968, se usa más con la intención de expresar un deseo que de describir la realidad predominante, en la que el software se construía con muy poca técnica y con unos lenguajes completamente inseguros.

Tony Hoare contribuyó en gran medida a cambiar esta situación, actuando en varias direcciones: primeramente, criticó duramente lenguajes como PL/I y Algol-68 que el consideraba formaban parte del problema. En ese sentido, abandonó junto con otros colegas el grupo IFIP 2.1 ocupado en la definición de este lenguaje Algol-68, y se unió al grupo 2.3 que había de definir buenas metodologías de programación. En su opinión, los lenguajes debían venir *después* y no antes de las metodologías.

En segundo lugar, fue cofundador del movimiento llamado de la "programación estructurada", editor del libro original con

dicho nombre y autor de uno de sus tres capítulos. Este libro contenía las semillas cuyos frutos disfrutamos hoy y que todavía son parte esencial de las metodologías actuales: la necesidad de razonar formalmente sobre los programas, la necesidad de disminuir el número de detalles a tener en cuenta simultáneamente, la técnica llamada de *diseño descendente*, la necesidad de retrasar las decisiones de representación de las estructuras de datos, la peligrosidad de usar la instrucción **go-to**, y otras ideas que chocaban frontalmente con las prácticas del momento.

Finalmente, colaboró con su colega Nicolás Wirth en el desarrollo del lenguaje Pascal, el lenguaje de referencia de la programación estructurada, y proporcionaron una semántica formal para el mismo. Era la primera vez que a un lenguaje de programación se le daba un significado matemático.

Su trabajo de 1972 "*Proof of correctness of data representations*" extiende el razonamiento a las estructuras de datos que implementan un *tipo abstracto*, concepto este último en el origen de lo que hoy conocemos como *programación orientada a objetos*. Sus aportaciones de "*invariante de la representación*" y "*función de abstracción*" se enseñan actualmente en los cursos de estructuras de datos.

A partir de 1974, sus trabajos sobre *monitores* proporcionan un mecanismo elegante y fiable para gobernar la concurrencia en un solo computador, y para razonar sobre la corrección de este tipo de programas. Los programas concurrentes proporcionan la ilusión de que el computador realiza varias tareas simultáneamente y se trata de un tipo de programación mucho más difícil y propensa a errores que la secuencial. Gracias a los monitores de Hoare no serán ya posibles desastres como los del sistema OS/360 previamente citado.

Muy poco después, aparecen en el mercado los primeros microprocesadores y se hace evidente que la programación concurrente va a convertirse enseguida en *programación distribuida* en la que intervendrán muchas máquinas simultáneamente. Con ello, los mecanismos preexistentes, incluidos sus monitores,

que suponen la existencia de una memoria común, van a quedar obsoletos. El es el primero en proponer un mecanismo, sus *Procesos Secuenciales Comunicantes* (CSP) de 1978, para domesticar el nuevo monstruo. La compañía INMOS, que fabrica redes de microprocesadores, se inspira en ellos y le pide su colaboración para desarrollar el lenguaje Occam. Se trata de un bello ejemplo de transferencia entre la universidad y la industria y de cómo las buenas ideas se pueden transformar en muy poco tiempo en productos industriales útiles para la sociedad.

Convierte a continuación su lenguaje CSP en un modelo algebraico de procesos y se preocupa por encontrar para este modelo operacional una semántica denotacional al estilo de las de su predecesor en Oxford, Christopher Strachey. Con ello extiende la capacidad de razonamiento formal a los programas concurrentes distribuidos. Su libro de 1985 "Procesos Secuenciales Comunicantes" puso todo este material en forma didáctica y ha sido libro de referencia en numerosos estudios de Posgrado.

Por todos estos méritos, en 1980 recibió el Premio Turing de la ACM (*Association for Computer Machinery*), el máximo galardón que un informático puede obtener, considerado en nuestro ámbito como el equivalente al Premio Nobel. En la mención del premio se indica que "su trabajo se caracteriza por una combinación inusual de profundidad, originalidad, elegancia e impacto". Se da un Premio Turing por año a la persona o personas que más han contribuido con su investigación al avance de nuestra ciencia. Precisamente en 2012 se celebró en todo el mundo el Año Turing, centenario del nacimiento de Alan Turing, considerado como el fundador de los computadores y de la ciencia informática.

Para la Universidad Complutense es un gran honor conceder la mención de Dr. Honoris Causa a un científico como Vd. que ha sido distinguido con el premio Turing. Estamos muy orgullosos de recibirle en nuestra comunidad como colega.

El Premio Turing no supuso para Vd. un freno en realizar buena investigación. A partir de 1994 Vd. se interesa por elaborar teorías unificadoras de la programación que engloben como casos particulares muchas de las teorías especializadas que utilizamos en la actualidad. Se trata de un objetivo de largo alcance al que todavía se sigue dedicando en el presente. A este respecto dedica numerosos trabajos, y un libro publicado conjuntamente con su colega He-Jifeng en 1998. En su visión, una ciencia madura debe disponer de esas teorías unificadoras. No hay nada equivalente en el campo de la programación al papel que juega la Mecánica Cuántica en la Física o el Álgebra en las Matemáticas.

En el primer caso, la teoría explica como casos particulares las teorías preexistentes sobre el calor, el electromagnetismo y la óptica. En el segundo, la teoría subsume las propiedades de muchas clases de números y de estructuras matemáticas previas a su aparición. Más importante aún es que dichas teorías unificadoras son capaces de predecir la aparición de nuevos fenómenos. Es el caso del recientemente descubierto bosón de Higgs, que fue predicho por una teoría unificadora de la Física elaborada cincuenta años atrás.

En su visión, una teoría unificadora de la programación ha de basarse en predicados y relaciones y tener un "sabor" algebraico. De hecho, Vd. ya nos ha proporcionado algunas de sus leyes. El objetivo final de esta teoría es alcanzar la corrección total de los programas que construimos los humanos, y que actualmente están tan plagados de errores.

En 2003 Vd. lanzó un *Gran Desafío* a la comunidad científica: profundizar en las teorías ya existentes, y crear las que fueran necesarias, para llegar a una meta difícil, pero en su opinión alcanzable, el llamado *Compilador-Verificador*. Si se alcanzara dicha meta, la creación de programas sería una tarea mucho más profesional y científicamente fundada de lo que es actualmente. La interacción de los ingenieros con dicha herramienta crearía programas correctos por construcción y las pruebas de ejecución tan solo confirmarían dicha corrección, un ideal que hoy nos parece todavía lejano.

Cada artículo suyo se convierte inmediatamente en un clásico y es ampliamente citado por trabajos posteriores de otros investigadores. En una entrevista que concedió recientemente a un periódico español, Vd. nos alertó del peligro de medir la capacidad científica de los investigadores por el *número* de artículos que publican cada año. Esta práctica, extendida por las agencias de evaluación y universidades del todo el mundo incluida España, es muy perniciosa para la creación de buena ciencia. Estimula la fragmentación innecesaria de los trabajos, la repetición parcial de los mismos, y el no realizar suficiente experimentación antes de enviar un trabajo a publicar. Algunos de sus artículos más citados le han supuesto a Vd. más de dos años de investigación. Seguramente una agencia de evaluación de nuestro país le reconveniría por su baja productividad.

Otro dato que revela su exquisito cuidado por el trabajo bien hecho, es que Vd. retiró un artículo de la revista *Comunicaciones de la ACM*, una vez que este había sido aceptado por el Comité Editorial, porque Vd. no estaba del todo satisfecho y había descubierto una forma de mejorarlo. Para que la audiencia pueda valorar la importancia de este gesto, diré que la revista *Comunicaciones de la ACM* es una de las más prestigiosas de nues-

tro campo. Muy pocos investigadores, españoles o extranjeros, consiguen publicar en ella, y se considera un gran mérito científico tener un artículo publicado allí.

Vd. fue nombrado miembro de la Royal Society en 1982, siguiendo los pasos de Alan Turing, el padre de la Computación. Ha recibido numerosas distinciones, como la AFIP Harry Goode Medal, el Premio Kioto de Ciencias de la Información, la IEEE John von Neumann Medal, el ACM SIGPLAN Distinguished Achievement Award, así como diez Doctorados Honoris Causa en diferentes países. En 2000 fue nombrado Sir por la Reina Británica por sus servicios a la educación y a las ciencias de la computación. Este título lo ostentaban muy pocos científicos antes que Vd. en el Reino Unido. Entre ellos cabe destacar a Sir Isaac Newton, padre de la Mecánica. Vd. podría con toda justicia ser calificado como el padre de la Programación científicamente fundada.

La relación de Vd. con nuestra universidad se establece sobre todo a partir de su obra científica. Es difícil encontrar un investigador de los departamentos de la Facultad de Informática de la UCM que no conozca sus trabajos. En particular, el contenido de sus publicaciones sobre verificación de programas y sobre concurrencia forma parte de las enseñanzas que se imparten en los grados y másteres de la Facultad de Informática de la UCM. A nivel de investigación, se han defendido en los últimos años varias tesis doctorales que utilizan o extienden sus teorías, tanto en el ámbito de los modelos concurrentes como en el de la verificación y certificación de programas.

A nivel más personal, no ha sido infrecuente coincidir con Vd. en algunos congresos y escuelas de verano. En particular, hemos enviado a la Escuela de Verano de Marktoberdorf, que Vd. ha dirigido durante más de veinte años, a docenas de nuestros graduados. Vd. fue conferenciante invitado en el CEDI (Conferencia Española de Informática) en Granada en 2005, al cual asistieron cerca de 2.000 investigadores. El contacto más reciente se ha producido el mes de Junio pasado, debido a su participación como conferenciante invitado en el congreso *Mathematics of Program Construction*, celebrado precisamente en esta Facultad.

Decía Séneca que "conceder un beneficio a un hombre de honores, en parte, recibirlo". Para mi universidad, para mi facultad, y para mi personalmente, es un gran honor y un valioso regalo otorgarle a Vd. este Doctorado *Honoris Causa* por la Universidad Complutense de Madrid, y lo es más aún el hecho de que Vd. haya aceptado recibirlo. Sir Tony Hoare, le damos las gracias por sus magníficas e inspiradoras contribuciones a la ciencia de la programación.

Muchas gracias.