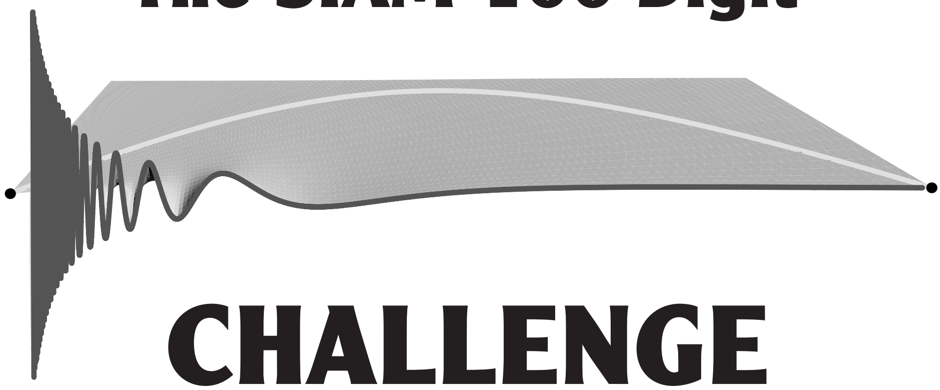
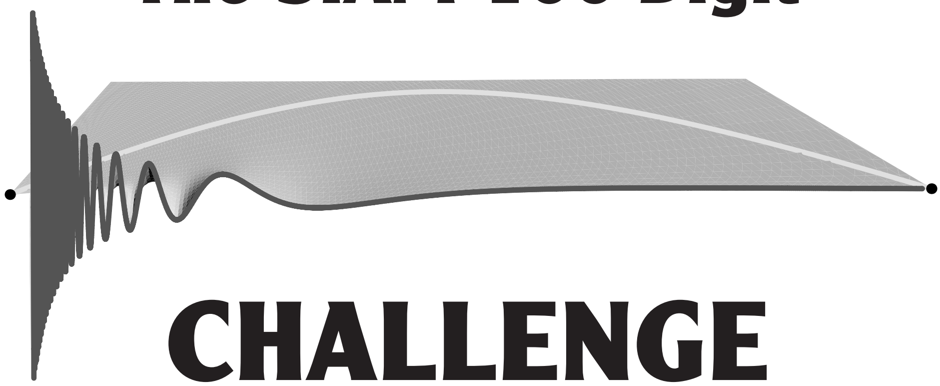
The SIAM 100-Digit



CHALLENGE

The SIAM 100-Digit



CHALLENGE

A Study in High-Accuracy Numerical Computing

Folkmar Bornemann
Technische Universität München
Munich, Germany

Dirk Laurie
University of Stellenbosch
Stellenbosch, South Africa

Stan Wagon
Macalester College
St. Paul, Minnesota

Jörg Waldvogel
Swiss Federal Institute of Technology (ETH) Zurich
Zurich, Switzerland

With a Foreword by David H. Bailey

siam
Society for Industrial and Applied Mathematics
Philadelphia

10 9 8 7 6 5 4 3 2

Library of Congress Cataloging-in-Publication Data

The SIAM 100-digit challenge : a study in high-accuracy numerical computing / Folkmar Bornemann … [et al.].
p. cm.
Includes bibliographical references and index.
ISBN 0-89871-561-X (pbk.)
1. Numerical analysis. I. Bornemann, Folkmar, 1967-
QA297.S4782 2004
518—dc22 2004049139

Contents

Foreword

Everyone loves a contest. I still recall when, in January/February 2002, I first read of the SIAM 100-Digit Challenge in *SIAM News*. Nick Trefethen's short article[1] introduced the 10 problems, then closed with the comment, "Hint: They're hard! If anyone gets 50 digits in total, I will be impressed." To an incorrigible computational mathematician like me, that was one giant red flag, an irresistible temptation. As it turned out I did submit an entry, in partnership with two other colleagues, but we failed to get the correct answer on at least 1 of the 10 problems and thus did not receive any award. But it was still a very engaging and enjoyable exercise.

This book shows in detail how each of these problems can be solved, as described by four authors who, unlike myself, belonged to winning teams who successfully solved all 10 problems. Even better, the book presents multiple approaches to the solution for each problem, including schemes that can be scaled to provide thousand-digit accuracy if required and can solve even larger related problems. In the process, the authors visit just about every major technique of modern numerical analysis: matrix computation, numerical quadrature, limit extrapolation, error control, interval arithmetic, contour integration, iterative linear methods, global optimization, high-precision arithmetic, evolutionary algorithms, eigenvalue methods, and many more (the list goes on and on).

The resulting work is destined to be a classic of modern computational science—a gourmet feast in 10 courses. More generally, this book provides a compelling answer to the question, "What is numerical analysis?" In this book we see that numerical analysis is much more than a collection of Victorian maxims on why we need to be careful about numerical round-off error. We instead see first hand how the field encompasses a large and growing body of clever algorithms and mathematical machinery devoted to efficient computation. As Nick Trefethen once observed [Tre98], "If rounding errors vanished, 95% of numerical analysis would remain."

As noted above, the authors of this book describe techniques that in many cases can be extended to compute numerical answers to the 10 problems to an accuracy of thousands of digits. Some may question why anyone would care about such prodigious precision, when in the "real" physical world, hardly any quantities are known to an accuracy beyond about 12 decimal digits. For instance, a value of π correct to 20 decimal digits would suffice to calculate the circumference of a circle around the sun at the orbit of the earth to within the width of an atom. So why should anyone care about finding any answers to 10,000-digit accuracy?

[1] See p. 1 for the full text.

In fact, recent work in experimental mathematics has provided an important venue where numerical results are needed to very high numerical precision, in some cases to thousands of decimal digits. In particular, precision on this scale is often required when applying integer relation algorithms[2] to discover new mathematical identities. An integer relation algorithm is an algorithm that, given n real numbers $(x_i,\ 1 \leqslant i \leqslant n)$, in the form of high-precision floating-point numerical values, produces n integers, not all zero, such that $a_1x_1 + a_2x_2 + \cdots + a_nx_n = 0$.

The best-known example of this sort is a new formula for π that was discovered in 1995:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k}\left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6}\right).$$

This formula was found by a computer program implementing the PSLQ integer relation algorithm, using (in this case) a numerical precision of approximately 200 digits. This computation also required, as an input real vector, more than 25 mathematical constants, each computed to 200-digit accuracy. The mathematical significance of this particular formula is that it permits one to directly calculate binary or hexadecimal digits of π beginning at any arbitrary position, using an algorithm that is very simple, it requires almost no memory, and does not require multiple-precision arithmetic [BBP97, AW97, BB04, BBG04]. Since 1996, numerous additional formulas of this type have been found, including several formulas that arise in quantum field theory [Bai00].

It's worth emphasizing that a wide range of algorithms from numerical analysis come into play in experimental mathematics, including more than a few of the algorithms described in this book. Numerical quadrature (i.e., numerical evaluation of definite integrals), series evaluation, and limit evaluation, each performed to very high precision, are particularly important. These algorithms require multiple-precision arithmetic, of course, but often also involve significant symbolic manipulation and considerable mathematical cleverness as well.

In short, most, if not all, of the techniques described in this book have applications far beyond the classical disciplines of applied mathematics that have long been the mainstay of numerical analysis. They are well worth learning, and in many cases, rather fun to work with. Savor every bite of this 10-course feast.

David H. Bailey
Chief Technologist
Computational Research Department
Lawrence Berkeley National Laboratory, USA

[2] Such algorithms were ranked among *The Top 10*—assembled by Jack Dongarra and Francis Sullivan [DS00]—of algorithms "with the greatest influence on the development and practice of science and engineering in the 20th century."

Preface

This book will take you on a thrilling tour of some of the most important and powerful areas of contemporary numerical mathematics. A first unusual feature is that the tour is organized by problems, not methods: it is extremely valuable to realize that numerical problems often yield to a wide variety of methods. For example, we solve a random-walk problem (Chapter 6) by several different techniques, such as large-scale linear algebra, a three-term recursion obtained by symbolic computations, elliptic integrals and the arithmetic-geometric mean, and Fourier analysis. We do so in IEEE arithmetic to full accuracy and, at the extreme, in high-precision arithmetic to get 10,000 digits.

A second unusual feature is that we very carefully try to justify the validity of every single digit of a numerical answer, using methods ranging from carefully designed computer experiments and a posteriori error estimates to computer-assisted proofs based on interval arithmetic. In the real world, the first two methods are usually adequate and give the desired confidence in the answer. Interval methods, while nicely rigorous, would most often not provide any additional benefit. Yet it sometimes happens that one of the best approaches to a problem is one that provides proof along the way (this occurs in Chapter 4), a point that has considerable mathematical interest.

A main theme of the book is that there are usually two options for solving a numerical problem: either use a brute force method running overnight and unsupervised on a high-performance workstation with lots of memory, or spend your days thinking harder, with the help of mathematical theory and a good library, in the hope of coming up with a clever method that will solve the problem in less than a second on common hardware. Of course, in practice these two options of attacking a problem will scale differently with problem size and difficulty, and your choice will depend on such resources as your time, interest, and knowledge and the computer power at your disposal. One noteworthy case, where a detour guided by theory leads to an approach that is ultimately much more efficient than the direct path, is illustrated on the cover of this book. That diagram (taken from Chapter 1) illustrates that many problems about real numbers can be made much, much easier by stepping outside the real axis and taking a route through the complex plane.

The waypoints of our tour are the 10 problems published in the January/February 2002 issue of *SIAM News* by Nick Trefethen of Oxford University as an intriguing computing challenge to the mathematical public. The answer to each problem is a real number; entrants had to compute several digits of the answer. Scoring was simple: 1 point per digit, up to a maximum of 10 per problem. Thus a perfect score would be 100. When the dust settled several months later, entries had been received from 94 teams in 25 countries. Twenty of those teams achieved a perfect score of 100 and 5 others got 99 points. The whole fascinating story, including the names of the winners, is told in our introductory chapter, "The Story."

The contest, now known as the *SIAM 100-Digit Challenge*, was noteworthy for several reasons. The problems were quite diverse, so while an expert in a particular field might have little trouble with one or two of them, he or she would have to invest a lot of time to learn enough about the other problems to solve them. Only digits were required, no proofs: not of the existence and uniqueness of the answer, not of the convergence of the method, not of the correctness of the result. Nevertheless, a serious team would want to put some effort into theoretical investigations. The impact of modern software on these sorts of problems is immense, and it is very useful to try all the major software tools on these problems so as to learn their strengths and their limitations.

This book is written at a level suitable for beginning graduate students and could serve as a text for a seminar or as a source for projects. Indeed, these problems were originally assigned in a first-year graduate course at Oxford University, where they were used to challenge students to think beyond the basic numerical techniques. We have tried to show the diversity of mathematical and algorithmic tools that might come into play when faced with these, and similar, numerical challenges, such as:

- large-scale linear algebra
- computational complex analysis
- special functions and the arithmetic-geometric mean
- Fourier analysis
- asymptotic expansions
- convergence acceleration
- discretizations that converge exponentially fast
- symbolic computing
- global optimization
- Monte Carlo and evolutionary algorithms
- chaos and shadowing
- stability and accuracy
- a priori and a posteriori error analysis
- high-precision, significance, and interval arithmetic

We hope to encourage the reader to take a broad view of mathematics, since one moral of this contest is that overspecialization will provide too narrow a view for one with a serious interest in computation.

The chapters on the 10 problems may be read independently. Because convergence acceleration plays an important role in many of the problems, we have included a discussion of the basic methods in Appendix A. In Appendix B we summarize our efforts in calculating the solutions to extremely high accuracies. Appendix C contains code that solves the 10 problems in a variety of computing environments. We have also included in Appendix D a sampling of additional problems that will serve as interesting challenges for readers who have mastered some of the techniques in the book.

All code in the book as well as some additional material related to this project can be found at an accompanying web page:

`www.siam.org/books/100digitchallenge`

We four authors, from four countries and three continents, did not know each other prior to the contest and came together via e-mail to propose and write this book. It took thousands of e-mail messages and exchanges of files, code, and data. This collaboration has been an unexpected benefit of our participation in the *SIAM 100-Digit Challenge.*

Notation and Terminology. When two real numbers are said to agree to d digits, one must be clear on what is meant. In this book we ignore rounding and consider it a problem of strings: the two strings one gets by truncating to the first d significant digits have to be identical. We use the symbol $\doteq$ to denote this type of agreement to all digits shown, as in $\pi \doteq 3.1415$.

When intervals of nearby numbers occur in the book, we use the notation 1.2345^{89}_{67} to denote $[1.234567, 1.234589]$.

Acknowledgments. First, we owe a special debt of thanks to the late John Boersma (1937–2004), who looked over all the chapters with an expert eye and brought to our attention many places where the exposition or the mathematics could be improved.

We are grateful to Nick Trefethen for his encouragement and advice, and to many mathematicians and members of other teams who shared their solutions and insights with us. In particular, we thank Paul Abbott, Claude Brezinski, Brett Champion, George Corliss, Jean-Guillaume Dumas, Peter Gaffney, Yifan Hu, Rob Knapp, Andreas Knauf, Danny Lichtblau, Weldon Lodwick, Oleg Marichev, Fred Simons, Rolf Strebel, John Sullivan, Serge Tabachnikov, and Michael Trott.

As long as a method of solution has been used by several contestants or can be found in the existing literature, we will refrain from giving credit to individuals. We do not suggest that any specific idea originates with us, even though there are many such ideas to be found in the book. Although each chapter bears the name of the author who actually wrote it, in every case there have been substantial contributions from the other authors.

Folkmar Bornemann, Technische Universität München, Germany

Dirk Laurie, University of Stellenbosch, South Africa

Stan Wagon, Macalester College, St. Paul, USA

Jörg Waldvogel, ETH Zürich, Switzerland

The Story

Problems worthy of attack
Prove their worth by hitting back
—Piet Hein

The 100-Digit Challenge began in 2001 when Lloyd N. Trefethen of Oxford University approached SIAM with the idea of publishing 10 challenging problems in numerical computing. SIAM liked the idea and the contest was officially launched in the January/February 2002 issue of *SIAM News*. Here is the full text of Trefethen's challenge:

> Each October, a few new graduate students arrive in Oxford to begin research for a doctorate in numerical analysis. In their first term, working in pairs, they take an informal course called the "Problem Solving Squad." Each week for six weeks, I give them a problem, stated in a sentence or two, whose answer is a single real number. Their mission is to compute that number to as many digits of precision as they can.
>
> Ten of these problems appear below. I would like to offer them as a challenge to the SIAM community. Can you solve them?
>
> I will give \$100 to the individual or team that delivers to me the most accurate set of numerical answers to these problems before May 20, 2002. With your solutions, send in a few sentences or programs or plots so I can tell how you got them. Scoring will be simple: You get a point for each correct digit, up to ten for each problem, so the maximum score is 100 points.
>
> Fine print? You are free to get ideas and advice from friends and literature far and wide, but any team that enters the contest should have no more than half a dozen core members. Contestants must assure me that they have received no help from students at Oxford or anyone else who has already seen these problems.
>
> Hint: They're hard! If anyone gets 50 digits in total, I will be impressed. The ten magic numbers will be published in the July/August issue of *SIAM News*, together with the names of winners and strong runners-up.
>
> — *Nick Trefethen, Oxford University*

At the deadline, entries had been submitted by 94 teams from 25 countries. The contestants came from throughout the world of pure and applied mathematics and included researchers at famous universities, high school teachers, and college students. The work was certainly easy to grade, and the results were announced shortly after the deadline: there were 20 teams with perfect scores of 100 and 5 more teams with scores of 99.

Trefethen published a detailed report [Tre02] of the event in the July/August issue of *SIAM News*, and his report nicely conveys the joy that the participants had while working on these problems. During the few months after the contest, many of the teams posted their solutions on the web; URLs for these pages can be found at the web page for this book.

Joseph Keller of Stanford University published an interesting letter in the December 2002 issue of *SIAM News*, where he raised the following point:

> I found it surprising that no proof of the correctness of the answers was given. Omitting such proofs is the accepted procedure in scientific computing. However, in a contest for calculating precise digits, one might have hoped for more.

This is indeed an important issue, one that has guided much of our work in this book. We have addressed it by providing proof for most of the problems and describing the large amount of evidence for the others, evidence that really removes any doubt (but is, admittedly, short of proof). Several responses to Keller's letter appeared in the January/February 2003 issue of *SIAM News*.

The Winners

Twenty teams scored 100 points and were deemed *First Prize Winners*:

— Paul Abbott, University of Western Australia, Nedands, and Brett Champion, Yufing Hu, Danny Lichtblau, and Michael Trott, Wolfram Research, Inc., Champaign, Illinois, USA

— Bernard Beard, Christian Brothers University in Memphis, Tennessee, USA, and Marijke van Gans, Isle of Bute, and Brian Medley, Wigan, United Kingdom ("The CompuServe SCIMATH Forum Team")

— John Boersma, Jos K. M. Jansen, Fred H. Simons, and Fred W. Steutel, Eindhoven University of Technology, the Netherlands

— Folkmar Bornemann, Technische Universität München, Germany

— Carl DeVore, Toby Driscoll, Eli Faulkner, Jon Leighton, Sven Reichard, and Lou Rossi, University of Delaware, Newark, USA

— Eric Dussaud, Chris Husband, Hoang Nguyen, Daniel Reynolds, and Christian Stolk, Rice University, Houston, Texas, USA

— Martin Gander, Felix Kwok, Sebastien Loisel, Nilima Nigam, and Paul Tupper, McGill University, Montreal, Canada

— Gaston Gonnet, ETH Zurich, Switzerland, and Robert Israel, University of British Columbia, Vancouver, Canada

— Thomas Grund, Technical University of Chemnitz, Germany

— Jingfang Huang, Michael Minion, and Michael Taylor, University of North Carolina, Chapel Hill, USA

— Glenn Ierley, Stefan L. Smith, and Robert Parker, University of California, San Diego, USA

— Danny Kaplan and Stan Wagon, Macalester College, St. Paul, Minnesota, USA

— Gerhard Kirchner, Alexander Ostermann, Mechthild Thalhammer, and Peter Wagner, University of Innsbruck, Austria

— Gerd Kunert and Ulf Kähler, Technical University of Chemnitz, Germany

— Dirk Laurie, University of Stellenbosch, South Africa

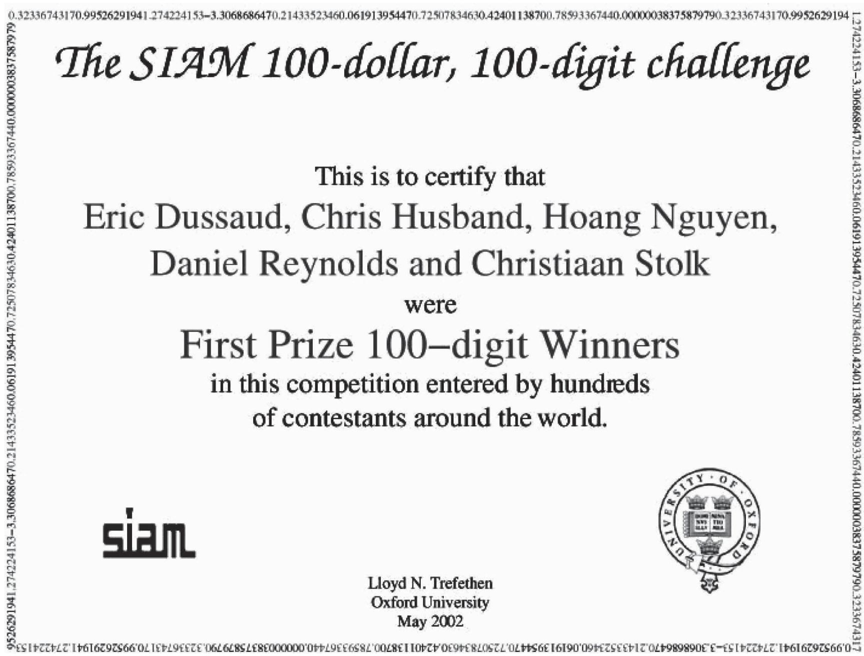


Figure 1 *Each winning team received $100 and an attractive certificate.*

- Kim McInturff, Raytheon Corp., Goleta, California, USA, and Peter S. Simon, Space Systems/Loral, Palo Alto, California, USA
- Peter Robinson, Quintessa Ltd., Henley-on-Thames, United Kingdom
- Rolf Strebel and Oscar Chinellato, ETH Zurich, Switzerland
- Ruud van Damme, Bernard Geurts, and Bert Jagers, University of Twente, Netherlands
- Eddy van de Wetering, Princeton, New Jersey, USA

Five teams scored 99 points and were counted as *Second Prize Winners:*

- Niclas Carlsson, Åbo Akademi University, Finland
- Katherine Hegewisch and Dirk Robinson, Washington State University, Pullman, USA
- Michel Kern, INRIA Rocquencourt, France
- David Smith, Loyola Marymount University, Los Angeles, California, USA
- Craig Wiegert, University of Chicago, Illinois, USA

Interview with Lloyd N. Trefethen

Many aspects of the story had been known only by Trefethen himself. We are grateful that he agreed to share his views about the contest and on general issues related to numerical computing in the following interview.

Lloyd Nicholas Trefethen is Professor of Numerical Analysis at Oxford University, a fellow of Balliol College, and head of the Numerical Analysis Group at Oxford. He was born in the United States and studied at Harvard and Stanford. He then held positions at the Courant Institute, MIT, and Cornell before taking the chair at Oxford in 1997. In 1985 he was awarded the first Fox Prize in Numerical Analysis.

Trefethen's writings on numerical analysis and applied mathematics have been influential and include five books and over 100 papers. This work spans the field of theoretical and practical numerical analysis. Of more general interest are his essays, such as "The Definition of Numerical Analysis" [TB97, App.], "Maxims about Numerical Mathematics, Computers, Science, and Life" [Tre98], and "Predictions for Scientific Computing 50 years from Now" [Tre00].

You received entries from 94 teams in 25 countries, from Chile to Canada and South Africa to Finland. How many people took part?

There were teams of every size up to the maximum permitted of six—altogether, about 180 contestants. Most of these scored 40 points or more. Of course, those are just the ones I know about! There were certainly others who tried some of the problems and didn't tell me. I would hear rumors that so-and-so had been staying up late nights. Hadn't he sent me any numbers?

Were there countries that you expected entries from, but got none?

No, I wouldn't say so. I got entries from six English-speaking countries, and I did not expect to hear much from the non-English world, where *SIAM News* is less widely distributed. So it was a nice surprise to receive entries from countries all over the world, including China, Russia, Spain, Slovenia, Greece, Argentina, Mexico, and Israel.

What kind of background and training did the contestants have? Was there a significant difference between those who succeeded and those who did not?

We had contestants of every sort, from amateurs and students to world-leading mathematicians. But it has to be admitted that overall, most teams were based at universities, and the higher scoring ones usually included an expert in numerical computation with a Ph.D. A crucial ingredient of success was collaboration. More than half of our teams consisted of a single person, but only five of those singletons ended up among the winners.

Wolfram Research (Mathematica®) had a winning team and Gaston Gonnet (one of the founders of Maple®) was part of a winning team. Were there teams from other mathematical software companies?

Cleve Moler, the creator of Matlab® and Senior Scientist at MathWorks, was teaching the undergraduate course CS 138 at Stanford at the time of the Challenge, and he put problem 1 on the students' take-home final exam in March 2002. I don't know how the students did, but I can assure you that Cleve got all the digits right.

Mathematica, Maple, and Matlab were used by many teams. What other software was used, and were there any unexpected choices?

Yes, the three M's were very popular indeed. We also had plenty of C and C++ and Fortran and many other systems, including Java™, Visual Basic®, Turbo-Pascal, GMP, GSL, Octave, and Pari/GP. One contestant attacked the problems in Excel, but he was not among the high scorers.

Did you know all the answers at the time you posed the questions?

I knew eight of them to the full 10 digits. My lacunae were the complex gamma function (Problem 5) and the photon bouncing between mirrors (Problem 2).

I'm afraid that my head was so swimming with digits in those last few weeks that I did something foolish. Shortly before the contest deadline I sent a reminder to NA Digest on the web to which I added a P.S. that I thought would amuse people:

```
Hint.  The correct digits are 56403899311367472691912742241578531
42257191239544746342078343703837583797932367495263306868621433534.
(Not in this order.)
```

As I said, at that time I didn't even know all 100 digits! To make my little joke I took the 90 or so digits I knew and padded them out at random. It never crossed my mind that people would take this clue seriously and try to check their answers with it. They did, and several of the eventual winners found that they seemed to have too many 4s and 7s, too few 2s and 5s. I am embarrassed to have made such a mistake and worried them.

Do you have an opinion on whether "digits" in a contest such as this should mean rounded or truncated?

What a mess of an issue—so empty of deep content and yet such a headache in practice! (Nick Higham treats these matters very well in his big book *Accuracy and Stability of Numerical Algorithms* [Hig96].) No, I have no opinion on what "digits" should mean, but I very much wish I had specified a precise rule when presenting the Challenge.

Why did the list of 100-point winners grow from 18 to 20 within a few days after you posted the results?

Mainly for reasons related to just this business of rounding vs. truncation. Would you believe that there were seven teams that scored 99 points out of 100? After I circulated a list with 18 winners, two of those seven persuaded me that I had misjudged one of their answers. Two more teams failed to persuade me.

For the teams that missed a perfect score by one or two digits, can you tell us which problems were missed? Did this coincide with your estimate of the difficulty ranking? Which problems did you think were most difficult?

The troublemakers were those same two I mentioned before: the complex gamma function and the photon bouncing between mirrors. The latter is a bit surprising, since it's not hard once you realize you need extended-precision arithmetic.

Are there any other facts from the results that help in the evaluation of the relative difficulty of the problems?

As I accumulated people's responses I quickly saw which problems gave the most trouble. Along with the gamma function, another tough one was the heating of a plate (Problem 8). It was also interesting to see for which problems some people were reporting lots of extra digits, like 50 or 500. A hard problem by this measure was the infinite matrix norm (Problem 3). When the contest ended I only knew about 18 digits of the answer, as compared with 50 digits or more for most of the other problems.

What experience (either before or after the contest) makes you believe that the problems were truly hard?

The problems originated in a problem-solving course that our incoming D.Phil. students in numerical analysis at Oxford take in their first term. (I got the idea from Don Knuth, and he told me once that he got it from Bob Floyd, who got it from George Forsythe, who got it from George Pólya.) Each week I give them a problem with no hints, and, working in pairs, they have to figure out as many digits of the answer as they can. In creating the problems I did my best to tune them to be challenging for this group of highly talented but not so experienced young people.

In your SIAM News report you indicated that perhaps some of the problems should have been harder. For example, you might have changed the target time in Problem 2 from 10 seconds to 100 seconds. Are there any other such variations to the other problems that might have made them more difficult? Or was it in fact a good thing that so many teams got all 100 digits?

I think five winning teams would have been better than twenty. As you suggest, some of the problems had a parameter I tuned to try to make them hard but not too hard. Changing a few numbers in the functions of Problems 1, 4, and 9, for example, could have made them even more devilish. On the whole I underestimated the difference between a pair of new graduate students with a week's deadline and a team of experienced mathematicians with months to play with.

What was your motivation in making the Challenge public? What did you expect to get out of it? Did you in fact get out of it what you wanted?

I love this kind of hands-on computing and I wanted others to have fun too. Too often numerical analysts get lost in theory and forget how satisfying it is to compute actual numbers.

What was SIAM's reaction when you first proposed the contest?

They liked the idea from the start. Gail Corbett, the *SIAM News* editor, is wonderfully encouraging of off-beat ideas.

You excluded people who might have seen these problems as members of the problem-solving squad at Oxford. Were all the problems taken from your course, and therefore already tested on students? Did you modify any for the Challenge or were they identical to the ones you give your students?

Yes, all the problems were taken from the course, and I changed nothing but a few words here and there. I didn't dare introduce untested problems and risk making an error.

What type of problem (whether from the Challenge or not) gives your students the most difficulty?

Students these days, alas, have often not had much exposure to complex analysis, so problems in the complex plane always prove hard. Personally, I don't see how one can go far in the mathematical sciences without complex variables, but there you are.

How many digits do they usually come up with?

On a typical meeting of our course there might be four teams presenting results at the whiteboard and they might get 4, 6, 8, and 10 digits. You'd be surprised how often the results appear in just that order, increasing monotonically! I guess the students who suspect their answers aren't so accurate volunteer fast, to get their presentations over with, and those who've done well are happy to wait to the end. One year we had an outstanding student who got nearly exact solutions two weeks running—we called him "Hundred Digit Hendrik."

How do you usually decide about the correctness of the digits? How did you decide at the contest?

At home at Oxford, I have almost always computed more digits in advance than the students manage to get. For the Challenge, however, scoring was much easier. In the weeks leading up to the deadline I accumulated a big file of the teams' numerical answers to the 10 problems. I'd record all the digits they claimed were correct, one after another. Here's an extract from Problem 8 with the names changed:

```
Argonne     0.4235
Berlin      0.4240113870 3
Blanc       0.2602370772 04
Blogg       0.4240113870
Cambridge   0.4240114
Cornell     0.4240113870 3368836379743366859326
CSIRO       0.3368831975
IBM         0.4240113870 336883
Jones       0.282674
Lausanne    0.42403
Newton      0.42401139
Philips     0.4240113870
Schmidt     0.4240074597 42
Schneider   0.4240113870 336883637974336685932564512478
Smith       0.4240113870 3369
Taylor      0.4240113870
```

In the face of 80 lines of data like this, based on different algorithms run in different languages on different computers in different countries, who could doubt that the correct first 10 digits must be 0.4240113870?

What is your opinion about Joe Keller's letter in SIAM News regarding the lack of proof of correctness, and the responses it got by some of the contestants?

Joe Keller has been one of my heroes ever since I took a course from him as a graduate student. Now, as I'm sure Joe knows, it would have killed the Challenge to demand that contestants supply proofs. But this doesn't mean I couldn't have said something in the write-up afterwards about the prospects for proofs and guaranteed accuracy in numerical computation. I wish I'd done that, for certainty is an ideal we must never lose sight of in any corner of mathematics. Those responses to Joe's letter "defended" me more vigorously than I would have defended myself.

Do you think the offer of money played any role in the interest generated? Is it correct that you originally only had a $100 budget in total?

Of course it did! Any whiff of money (or sex) makes us sit up and take notice. My "budget" consisted of the Trefethen family checkbook. But do you know something funny? After I'd sent certificates to the winners, I found that some of them seemed to care quite a bit about the money for its own sake, not just as a token, even if at best they could only hope to pocket a fifth or a sixth of $100. I guess there's a special feeling in winning cash that just can't be matched by certificates and publicity.

You decided to award three $100 prizes, but then an anonymous donor stepped forward and made it possible to give each team $100. Is the donor a mathematician? Do you know him or her personally? What especially impressed the donor about the contest?

What a thrill it was one day to receive his offer out of the blue! Now that the contest is well past, the donor does not object to having his name revealed. He is William Browning, founder and President of Applied Mathematics, Inc. in Gales Ferry, Connecticut. I haven't yet met Dr. Browning, but I know from our email exchanges that he was happy to support our field in this way. In one message he wrote:

> I agree with you regarding the satisfaction—and the importance—of actually computing some numbers. I can't tell you how often I see time and money wasted because someone didn't bother to "run the numbers."

Were you surprised by the response, and if so, why?

I was surprised by people's tenacity, their determination to get all 100 digits. I had imagined that a typical contestant would spend a dozen hours on the Challenge and solve three or four problems. Well, maybe some of them started out with that plan. But then they got hooked.

How much email traffic did the contest generate?

Megabytes! Those 94 entries corresponded to about 500 email messages. People would send results in the form of 30-page documents, then send updates and improvements, ask me questions, and so on. For some weeks before the deadline I seemed to be spending all my time on this. I didn't plan for that, but it was great fun.

Were there any WWW groups formed that openly discussed the problems? Did you scan the net for those and ask them to keep quiet?

I didn't scan the net systematically, but I heard about a group in Germany that was circulating ideas on the web. I asked them to go private, which they did.

How important was the Internet for the contest? Did people find ideas, software, etc. there?

It was crucial in spreading word of the event, and in the case of the SCIMATH team of three, the participants found each other through the Internet and so far as I know have still never met face-to-face. I think the Internet helped many contestants in technical ways too. For young people these days it is simply a part of life. At Oxford we have one of the world's best numerical analysis libraries, but you don't often see the D.Phil. students in it.

Where, other than SIAM News and NA-Net, was the contest publicized?

There were half a dozen places. The two I was most aware of were Wolfram Research's MathWorld web site and a notice published in *Science* by Barry Cipra with the heading "Decimal Decathlon."

What was the biggest surprise for you on the human side of the story?

How addicted people got to these problems! I received numerous messages of thanks for the pleasure I had given people, and quite a few asked if I would be making this a regular event. (No, I won't.)

What was the biggest disappointment?

I wish there had been more entries from my adopted country of Britain.

What was the biggest surprise as to the solutions that were put forward?

One day I got a message telling me that Jean-Guillaume Dumas of the LinBox team had solved Problem 7 *exactly* with the help of 182 processors running for four days. He showed that the answer is a quotient of two 97,389-digit integers. Wow!

Did you learn any new mathematics in reviewing the solutions?

On every problem there were surprises and new ideas, for as this book shows, these problems have many links to other topics. But if I'm honest I must tell you that with a thousand pages of mathematics to evaluate in a matter of days, I didn't have time to learn many new things properly.

Some people might say that a computation to more than six significant digits is a waste of time as far as real-world applications go. What do you think?

I am very much interested in this question, and I have distilled some of my views into the notion of a *Ten-Digit Algorithm*: "Ten-digits, five seconds, and just one page." Some writings on ten-digit algorithms are in the pipeline, so if you'll forgive me, I won't elaborate here.

You said that round-off error plays a small role compared to algorithm design, and that feeling is reflected in your choice of problems, since only one required high precision. Can you expand on this point?

In my 1992 *SIAM News* essay "The Definition of Numerical Analysis" [TB97, App.] I argue that controlling rounding errors is just a small part of numerical analysis, maybe 5% or 10%. The main business of this field is the development of algorithms that converge fast, and the ideas behind these algorithms would be just as necessary even if computers could work in exact arithmetic. So yes, it seems fitting that only 1 of the 10 problems, the photon bouncing off mirrors, is one where you have to think carefully about rounding errors.

In your report in SIAM News you asked, perhaps tongue-in-cheek, whether these problems could be solved to 10,000 digits. Do you think that such work has value beyond mere digit-hunting?

Humans have always progressed by tackling challenges, whether real or artificial.

Contests like yours where a mathematician announces, "Look, here is a hard problem that I know how to solve. I wonder whether you are able to do it too," were quite common at the time of Fermat, the Bernoullis, and Euler. But we do not see many such challenges today, certainly not in numerical computing. Should there be more of them? Should a numerical analysis journal consider starting a problem section where problems in the style of the Challenge are posed?

That's a very interesting question, for you're right, there was a different style in the old days, when science was an activity of a small elite and had not been professionalized. Yes, I think it would be good to have more challenges nowadays, though journals might not be the right medium since they are so slow.

Are there any important morals to draw from the challenge?

Huckleberry Finn begins with some remarks on this subject.

What has been the impact of the Challenge?

As Mao Tse-Tung or Zhou En-Lai is supposed to have said when asked about the impact of the French Revolution, "It's too early to tell!"

What kind of activity developed after the contest? Do you know of any web pages, papers written, talks given?

Yes, there were at least a dozen talks and tech reports, for the contestants were wrapped up in these problems and wanted to share their good ideas. Of course, this book of yours is the most extraordinary result of the Challenge.

What was your reaction when you heard that a book was being planned?

I was amazed and delighted.

You have told us that the Challenge has shown us a world that seems as distant from von Neumann as from Gauss. What makes you say that?

I imagined that the Challenge would unearth 10-digit solutions that would be a kind of culmination of 50 years of progress in algorithms and software. What in fact happened seems more of a transcendence than a culmination. I think the Challenge and its amazing aftermath—the 10,000-digit mix of fast algorithms, symbolic computation, interval arithmetic, and global collaboration displayed in this book—show that we have entered a world that would be unrecognizable even to the giants of the recent past such as Turing, von Neumann, or Wilkinson. It is a world that seems hardly less distant from von Neumann, who knew about computers and jet aircraft, than from Gauss, who lived in the time of Napoleon.

Would you do it again?

Absolutely.

Comments by the Contestants

We solicited some comments from the prize-winning teams and include a summary here. The diversity of approaches is noteworthy. Several contestants programmed everything from scratch "for sport," but most teams used software packages at some point. It is clear from the responses that the primary motivation for the sometimes great effort required was simply the satisfaction one gets by solving a difficult problem. But the recognition was nice too, and so all participants, like us, are grateful to Trefethen for setting the challenge and publicizing the results.

Another general point that comes through is that problems that offered little in the way of analytical challenge were considered the least favorite. Yet all the problems can teach us something about numerical computation, and we hope that point comes through in this book.

Why do it?

— I thought it would be a good way to promote numerical analysis within the department. *(Driscoll)*

— I have done a lot of competitions before and like doing them. *(Loisel)*

— Once I read the problems over I was hooked and used my spare time until I cracked them all. *(van de Wetering)*

— The general nature of the challenge, and, for some problems, the desire to test out some new functionality in *Mathematica* that was under development. *(Lichtblau)*

— I was taking a numerical differential equations class taught by Dr. Rossi (who solved the random-walk problem). We were studying quadratures when the challenge was issued and Dr. Rossi offered an A to any student who could solve the quadrature problem to 10 digits. I couldn't. However, I did get interested in the challenge and began attending our weekly meetings. (*Faulkner)*

— My main motivation was that I have always been interested in problem solving. In this connection, I still regret that the Problems and Solutions section of *SIAM Review* was discontinued at the end of 1997, just at the moment when I went into an early retirement! The Problem section played an important role as a forum where scientists could bring their mathematical problems to the attention of a readership of widely varying expertise. I believe the editors of the journal underestimated the section's role as such a forum when they decided to drop it. All four members of our team are retired mathematicians from Eindhoven University of Technology. At the university we share a big office where we can meet regularly. After a month or so we had solved six problems and were considering submitting our solutions (recalling Trefethen's comment that he would be impressed "if anyone gets 50 digits in total"). Fortunately we did not do that but continued till we had solved all 10 problems. *(Boersma)*

Was the contest worth it?

— Yes, because of the perfect score and because the undergraduate on the team is now contemplating graduate study in numerical analysis. *(Driscoll)*

— For the satisfaction of solving the problems. *(van de Wetering)*

— It was fun and I learned quite a few things on the way. *(Kern)*

— We were able to demonstrate advantages in *Mathematica*'s arithmetic and also to show the utility of some of the sparse linear algebra under development at the time. *(Lichtblau)*

— The challenge completely redefined the way I viewed computational mathematics. At that time I had taken a course in numerical linear algebra and was taking a course in numerical DEs. I learned that just because you knew how to compute a QR decomposition didn't mean that you knew anything about numerical linear algebra. *(Faulkner)*

— Yes, definitely. It was great fun. The problems of the Challenge formed a nice mixture, to be solved by both analytical and numerical methods. As for me, I preferred the analytically oriented problems like Problems 1, 5, 6, 8, 9, and 10. Also, the use of a computer algebra package (*Mathematica* in our case) greatly contributed to the successful outcome. *(Boersma)*

What insight did you find to be most satisfying?

— I enjoyed making headway with nonnumerical (i.e., analytical) methods. It was great fun to find useful tricks and do convergence acceleration and error estimation. Also, there is something very satisfying about using independent methods and getting digits that match. *(van de Wetering)*

— The whole thing. Going from, "How can one do that?" when I first looked at the problems, to, "Yes, it can be done" at the end. *(Kern)*

— An integration by parts trick to speed the convergence for Problem 1. It turned out not to be the most elegant solution, but it blew the problem open for us. *(Driscoll)*

— I liked the infinite matrix norm problem. I wrote a routine that was able to compute Ax for any vector x without instantiating the matrix. This allowed me to use a $50{,}000 \times 50{,}000$ matrix in a shifted power method and still terminate in seconds. I also used a kind of hierarchical technique. I used the eigenvector for the $25{,}000 \times 25{,}000$ case as the initial vector in the $50{,}000 \times 50{,}000$ case, so very few iterations were needed. *(Loisel)*

— That I could come up with a nice solution for Problem 3, as opposed to doing it by a brute force extrapolation. In short, some thinking about this problem paid off. *(Strebel)*

— I really liked watching my professors struggle on these problems, just as we undergraduates struggle on the problems they give us. *(Faulkner)*

— I learned most from the solutions of Problems 6, 10, and 5. At first I was scared of Problems 6 and 10, because my knowledge of probability theory is rather superficial. I felt at ease again after I had found the right analytical expressions to manipulate. Steutel and I were able to generalize Problem 6 to that of a general biased two-dimensional random walk, and we even considered writing a paper on this generalized problem. However, in January 2003, Folkmar Bornemann brought to our attention that the generalized problem had been treated in papers by Henze [Hen61] and Barnett [Bar63]; so that was the end of our paper to appear! All this has now been incorporated, in a most satisfactory manner, in Chapter 6 of the present book. *(Boersma)*

Which problem was the hardest?

— Problem 5. Basically I brute-forced it and was sweating over the last digit or two. *(Driscoll)*

— Problem 5 was the one question where I had to do some research, having never tackled this type of problem. It was the last question that I tackled, so feeling I had 9 out of 10 provided enough motivation to sort this one out. *(Robinson)*

— Problems 3 and 5, the latter only because we failed to see the "correct" approach. *(Lichtblau)*

— Problem 5 was by far the hardest; it took Jos Jansen and me over a month to crack it. Our calculations led to maxima of $|f(z) - p(z)|$ which were decreasing, thus showing that we had not yet found the best polynomial. Finally we found in the literature a necessary and sufficient criterion for the polynomial $p(z)$ to be the polynomial of best approximation for $f(z)$ on $|z| = 1$. This criterion could be implemented into an algorithm that produced the best polynomial $p^*(z)$ and the associated maximum$|f(z) - p^*(z)|$ extremely rapidly. *(Boersma)*

Were any problems extremely easy for you?

— Problem 10. I'm pretty good with conformal mapping and knew right away that it was a Maple three-liner. *(Driscoll)*

Did you have a favorite problem? A nonfavorite?

— I'd say Problem 1. It's really not obvious that the limit exists. *(Kern)*

— In a strange way Problem 10 is my favorite, which is at its core not a numerical problem at all. Of course it's a physics problem. It also reminded me of the origins of Fourier theory. A fun fact is that the resulting answer is a very quickly converging series (you need only one term to get to the answer). *(van de Wetering)*

— Problems 4 and 9 provided little satisfaction. *(Robinson)*

— I liked Problem 6. It leads pretty quickly to some interesting numerics and some difficult theoretical questions. There are also effective means for solving it all along a full spectrum from pretty simple to sophisticated—direct simulation is about the only guaranteed failure. *(Driscoll)*

— My least favorite was Problem 2, as there is no clever way to solve it. *(Strebel)*

Were you confident in your digits?

— I had more than one method for most problems. But you're never 100% confident. There is always the chance of dropping a digit or misinterpreting a problem. *(van de Wetering)*

— I had either two methods to solve the problems or an independent check of the result, so I was reasonably confident. *(Strebel)*

— Problem 10 was frustrating. I felt that there must be a neat way of doing this, but could only come up with slowly converging double infinite series. In the end, I stuck with a brute force approach, but this was the question that I would have been least surprised to have got wrong. (*Robinson)*

Have you any comments on digits you missed?

— I missed the last digit for Problem 2. I must admit it took me a while to realize it was so hard. I started using the geometric modelling part of a mesh generator written by a colleague, but as it used single precision, I knew I had to go to Maple. I programmed the whole thing in Maple and explored the way the results depended on `Digits`. So I found out I needed a lot of digits, and I used `Digits := 40`. Except, for some reason, I set `Digits := 20` in the calculation I copied in my report, and there you are, last digit is wrong. *(Kern)*

Many of the teams were partially or entirely nonmathematicians. If you are not primarily a mathematician, do you feel that this sort of exercise is valuable to the applied math or scientific computation community?

— I think it is a valuable exercise for all applied sciences. It drives home the point that convergence doesn't always come easy and that when a black box spits an answer at you, you really need to look under the hood and get a clue if it was good/bad/completely wrong (which is very true in my line of work: finance). Furthermore, it is valuable to realize that solvable problems often allow for a wide variety of methods. *(van de Wetering)*

Did you find any unusual sources?

— Only for Problem 5 did I do some serious literature research. In this case, knowing that Lloyd N. Trefethen had worked in this area proved to be a disadvantage: the methods in his papers really can't produce anything close to ten digits. I wasted some hours barking up this tree. *(Driscoll)*

Have you any additional comments?

— It was a pleasure having the opportunity in my spare time to work on these challenging problems. The charm of these problems is that they cover a lot of ground, are mostly nontrivial but are expertly crafted to require little overhead: no specialized knowledge, software, or hardware is needed to crack them as posed. *(van de Wetering)*

— We had a gung-ho MATLABer (me) and a gung-ho Mapleist (DeVore). Our approaches to problems were, almost without exception, entirely unrelated. Though ultimately all the problems save Problem 2 were eminently doable in standard MATLAB, I picked up some extra respect for Maple. With regard to numbers 1 and 9 in particular, I would say that Maple can be devilishly difficult to coax into numerical integrals, but MATLAB's packaged offerings in this area aren't even on the map. *(Driscoll)*

— Overall I really enjoyed the experience and feel that I got to contribute a lot to the team. The University of Delaware Mathematics Department is a great place for undergrads, but I know that not all schools allow undergrads to be so involved. I am not a straight A student, but I could handle some of these problems, so if I have any advice to share, it is that professors should let their students get involved in learning the challenges of mathematics. You never know where the great ideas will come from. *(Faulkner)*

Relative Difficulty of the Problems

One way of evaluating the relative difficulty is to look at the statistics of how the teams performed on the problems. Table 1 shows that Problem 4 was clearly the easiest and Problem 5 the hardest. Looking at the number of teams that missed a perfect score of 10 by just 1 digit shows that Problems 1, 3, and 7 were difficult, while the mean values of the points obtained show that Problems 6 and 8 were tough. And Problem 10 was tried by the smallest number of teams, probably because of the technical difficulty of dealing with Brownian motion.

However, the general spread of results through the table is further evidence that the contest was well designed.

Table 1 *Number of teams getting k digits on problem j and the mean value of the points obtained for that problem.*

Correct digits:	0	1	2	3	4	5	6	7	8	9	10	Mean value
Problem 1 (87 teams)	5	2	1	4	4	–	3	2	–	11	55	8.2
Problem 2 (80 teams)	2	–	3	–	4	3	6	2	1	1	58	8.6
Problem 3 (78 teams)	1	1	6	1	–	–	1	–	2	15	51	8.8
Problem 4 (84 teams)	–	3	–	–	–	–	–	–	1	–	80	9.7
Problem 5 (69 teams)	5	7	5	6	5	1	1	1	1	15	22	6.3
Problem 6 (76 teams)	11	2	3	–	1	3	1	–	3	1	51	7.6
Problem 7 (78 teams)	–	1	4	1	5	–	–	–	1	14	52	8.8
Problem 8 (69 teams)	6	–	3	1	7	1	1	2	–	–	48	7.9
Problem 9 (80 teams)	3	2	3	1	2	1	1	4	6	4	53	8.4
Problem 10 (62 teams)	–	2	1	1	3	1	2	1	–	–	51	8.9