The Proceedings of the 20th Symposium on Computer Arithmetic are dedicated to these distinguished colleagues retiring recently or soon:

Mike Cowlishaw retired as IBM Fellow, the top technical position in IBM, in March 2010. He is also a Visiting Professor at the Department of Computer Science at the University of Warwick. Mike is known in the computer arithmetic community as the father of decimal floating-point arithmetic for his lead role in convincing the IEEE 754 standard revision committee to address the base ten radix.

Less known to the arithmetic community, but well known in other circles such as IBM programmers is Mike’s work in creating the programming language REXX. It is an interpreted programming language that can be used on PC’s to workstations to even mainframes. Most mainframe programmers as well as IBM’s engineers and programmers start their career learning REXX. Mike also helped in providing a syntax directed editor called, LEXX, for the new Oxford English Dictionary.

Mike started into decimal floating-point arithmetic when he advocated that Java adopt the BigDecimal class which eventually got accepted in JSR13 and now is included in Java 5.0. Mike also wrote the DecNumber package to show the utility of these formats on almost every platform from PC to mainframe.

He then got involved in the revision to the IEEE 754-1985 Binary Floating-point Standard and convinced the committee to standardize decimal floating-point formats. Mike invented the compression encoding of BCD digits into Densely Packed Decimal (DPD) format which he provided royalty-free. His intent was to get the committee to standardize the decimal encodings but instead he had an overwhelming response where the committee not only standardized the encodings but took on the task of studying and standardizing the underlying arithmetic of these decimal formats. Mike undertook some of the final editing of the IEEE 754-2008 Floating-point Standard that not only describes binary format, but due to Mike, includes decimal encodings and arithmetic. Without Mike decimal floating-point would never have been included in the IEEE 754-2008 standard.

http://speleotrove.com/mfc/mfc_biography.html
http://en.wikipedia.org/wiki/Mike_Cowlishaw
Peter Kornerup was born in Aarhus, Denmark, in 1939. After studies in Mathematics at Aarhus University, he was hired first as teaching assistant, then as "Amanuensis", and later on as associate professor at Aarhus. As a visiting associate professor in the University of Southwestern Louisiana, he started research and teaching in computer arithmetic and its language support. From April 1971 to December 1987, he was chairman of the computer science department of Aarhus University. In 1988 he became a full professor in Odense University, where he is an emeritus since July 2009.

Peter Kornerup has always been deeply involved in the ARITH community: he organized ARITH-6 in Aarhus in 1983, and served several times as co-program chair: for ARITH-10 in Grenoble in 1991; ARITH-14 in Adelaide in 1999; and for ARITH-18 in Montpellier in 2007. He also served as steering committee chair of the RNC series of conferences, and as co-guest editor of several special issues on computer arithmetic of IEEE Transactions on Computers and Theoretical Computer Science. He has always been interested in number systems and their properties: just to give a few examples, he invented the slash number systems, and introduced a Lexicographic Binary Representation of the Rationals with David Matula, he produced original and deep results on digit-set conversions, on radix representations of rings. His recent book with David Matula, Finite Precision Number Systems and Arithmetic (Cambridge University Press, 2010) is the result of more than 20 years of work, and is probably the best reference on number systems.

After this acknowledgement of the great achievements of the scientist, we must say something about the man: Peter is a very trustworthy, honest, modest and friendly person. It has always been a pleasure to meet him all these past years, and we hope we will have many opportunities to welcome him again in our future meetings.
Ulrich Kulisch was director of the Institute of Applied Mathematics at the University of Karlsruhe for over 30 years. Prof. Kulisch is well known in the area of floating-point arithmetic. But before his dedication to this subject he had performed other pioneer work. One example is the introduction of computer science teaching in 1968, which later led to the foundation of the Faculty of Informatics at the University of Karlsruhe, the first one in Germany. Later he was probably the first one who gave students in his programming classes access to a kind of personal computer, while batch processing of punch cards and punch tapes were standard. These were Z80 based systems running CP/M, and the Pascal compiler resided on a single floppy disk.

In his publications Prof. Kulisch laid the foundation for a theoretical approach to computer arithmetic by defining the mathematical concept of a semimorphism. One crucial point was the application of this theory not only to operations on real numbers, but also on vectors and matrices as well as on complex numbers. This led to requirements on the implementation of floating-point arithmetic on computers. Several software and hardware solutions were developed.

The usefulness of this theory is proven by the fact that algorithms exist which allow the computation of narrow bounds for the solution of mathematical problems in spite of the rounding errors inherent in floating-point arithmetic. Implementing these algorithms is facilitated by programming languages which allow easy access to the features of the floating-point arithmetic. Intervals for scalars, vectors, and matrices are intrinsic data types in these languages and arithmetic operations are defined for them. Thus Prof. Kulisch has always been interested in programming languages. Extensions to Pascal, Fortran, and C have been defined and implemented. Over time a rich library of programs has been built which deliver guaranteed bounds for solutions of a wide spectrum of mathematical problems.

Prof. Kulisch has always been successful in finding partners for his projects. He is easily able to convince an audience of the importance of doing mathematics on a computer in the right way. In his presentations there are always examples where standard methods for solving a mathematical problem on a computer deliver huge errors without any warning. Several collaborations with industry resulted from his activities. A long running cooperation was the development of ACRITH with IBM Böblingen. This was a Fortran based package for S/370 systems providing the base floating-point arithmetic along with a library of problem-solving functions. There was hardware assist available on several S/370 systems.

http://www.math.kit.edu/ianm2/~kulisch/media/curricengl1.pdf
Tomás Lang has a distinguished and long record of outstanding research contributions to computer arithmetic and computer architecture fields. His work includes seminal results on digit-recurrence algorithms and implementations for division and square root, fast left-to-right multiplication, various CORDIC computations, and floating-point arithmetic. He has also made contributions to the design of low-power arithmetic, online arithmetic, fast adders, systolic algorithms, and decimal arithmetic. His results have had a strong impact on research in arithmetic and on industrial practice. His contributions to this symposium and the IEEE Transactions on Computers have been extensive. In the computer architecture field he made notable contributions to research in vector multiprocessors, interconnection networks, scheduling, and memory models. Professor Lang has been a great mentor and a colleague: he worked with many of the most active members of the ARITH community and influenced their careers. He coauthored two textbooks on digital systems: Digital Systems and Hardware/Firmware Algorithms, Wiley 1985, and Introduction to Digital Systems, Wiley 1998. In the arithmetic field he coauthored a monograph Division and Square Root: Digit-Recurrence Algorithms and Implementations, Kluwer 1994, and a textbook Digital Arithmetic, Morgan-Kaufmann 2004. Tomás Lang received the BS degree in electrical engineering from the Universidad de Chile in 1965, the MS degree from the University of California, Berkeley, in 1966, and the PhD degree from Stanford University in 1974. He is a professor emeritus in the Department of Electrical Engineering and Computer Science at the University of California, Irvine. Previously, he was a professor in the Department of Computer Architecture at the Polytechnic University of Catalonia, Spain, and a faculty member in the Department of Computer Science at the University of California, Los Angeles.
Martin Schmookler is well known in the field of computer arithmetic. He has participated in the design of almost every IBM computer in the pSeries workstation product line and many in the zSeries mainframe product line. He began his career working on the IBM 7030 computer code named “Stretch” which was the forerunner to the System/360 computers and which recently celebrated its 50th anniversary. After 53 years with IBM, he is retiring from high performance computer design.

Marty is an innovator whose ideas are timeless. His ideas go back to the computers of the 1960’s and are still in use today. His design of a decimal adder was implemented on the IBM S/360 model 195 and on the IBM z900 over 30 years later. Marty was involved in specifying the Altivec/VMX design of Power series vector units. From his many years experience, he is the “go to person” to understand how to design a fused multiply-add dataflow as well as an expert in division and square root. He has taught many designers about the quirks of the multiplier’s sign-extension encoding carries interfering with the addend in an FMA dataflow. He relishes the opportunity to teach others from his experiences. He relates his conversations with all the researchers of counter tree reduction and will explain the principles of end-around-carry adders over lunch. Marty knows how to make these designs work and has written some of the first papers describing both his innovation and acquired knowledge, such as on how to support subnormal operands at speed.

Marty is a teacher of computer design. After 10 years with IBM, they paid for him to get his Ph.D. at Princeton University. Then in 1976, he took a one-year position as a visiting professor in the Computer Science department at the University of Texas in Austin. Marty knows the people behind each development in computers and he knows the unwritten ways of how to get a design to work. Marty is a teacher with over 50 years experience making floating-point units work. He is active in the computer arithmetic community as a many time member of the program committee for the IEEE Symposium on Computer Arithmetic.