Charles D. Allison

Curriculum Vitae

Education

- **M.S. (Ph.D. ABD)**   <u>Applied Mathematics</u>, *University of Arizona*   1985
  Emphasis: Computer Science
  Research: Automatic Stability Analysis of Matrix Algorithms
- **M.S.**   <u>Mathematics</u>, *Brigham Young University*   1978
  Minor: Statistics
  Thesis: "Floquet Theory via Functional Equations"
- **B.S.**   <u>Mathematics</u>, *Brigham Young University*   1976
  Minor: Portuguese

Awards

- *Faculty Excellence Award*, Utah Valley State College, 2004
- *Outstanding Educator of the Year*, Utah Valley State College, 2004
- *NASA Ph.D. Traineeship in Applied Mathematics*, Univ. of Arizona, 1981 – 1983

Memberships (past and present)

- Association of Computing Machinery
- Mathematical Association of America
- Sigma Xi, The Scientific Research Society
- Society for Industrial and Applied Mathematics
- American Mathematical Society
- Pi Mu Epsilon, The Honorary National Mathematics Society

Teaching Experience

Utah Valley University, Orem, UT

Associate Professor, Dept. of Computing & Networking Sciences, 2006 – present.
Assistant Professor, Dept. of Computing & Networking Sciences, 2001 – 2006.
> Teaching computation theory, discrete structures, C++ software development, numerical software, design patterns, analysis of programming languages, Java ▪ Faculty Senate President (2008-2010) ▪ Faculty senate representative (2004-2007) ▪ Program Coordinator of CS emphasis ▪ Member of Undergraduate Research Council ▪ Member of the Math Task Force Committee ▪ Member of CS Department Masters Degree Curriculum Committee ▪ Member of Faculty Workload Conference Committee.

Pima Community College, Tucson, AZ

Instructor, Department of Computer Science, 1983 – 1984.
> Taught FORTRAN, BASIC, Pascal, and Introduction to Computing ▪ Developed two new courses ▪ Appointed department chair after one year.

Adjunct Instructor, Department of Computer Science, 1983, 1984 – 1988.
    Developed and taught courses in intermediate C programming and relational databases.

University of Arizona, Tucson, AZ

Adjunct Instructor, Department of Computer Science, 1982
    Taught Advanced Scientific Programming.

California State University, Long Beach, CA

Adjunct Professor, Department of Mathematics, 1980 – 1981
    Taught Ordinary Differential Equations and Finite Mathematics.

Rancho Santiago College, Santa Ana, CA

Adjunct Instructor, Department of Mathematics
    Taught Numerical Methods.

Golden West College, Huntington Beach, CA

Adjunct Instructor, Department of Mathematics and Computer Science, 1978 – 1980
    Taught Beginning Algebra and APL Programming.

Brigham Young University, Provo UT

Instructor, Department of Mathematics, 1977 – 1978
    Taught College Algebra, Trigonometry and Linear Programming (full-time).

Industry Experience

2000 - 2001: Senior Software Engineer, Novell, Inc., Provo, UT
As a member of the Custom Development Group, performed all phases of software engineering for small projects to serve Novell's clients ▪ Projects involved Java, C++, and Internet technologies

1998 - 2000: Senior Technical Advisor, Ingenix, Inc., Salt Lake City, UT
Provided mentoring and systems architecture consulting to projects using C++ and Oracle for the medical billing industry ▪ Taught internal courses on C++ and software engineering ▪ Developed a tool to automate the use of exceptions and the use of tables as objects in Pro-C++ ▪ Developed a framework for automated unit testing of C, C++ and Java programs

1997 - present: President, Fresh Sources, Inc., Layton, UT
President of a company offering training, mentoring, and custom programming services in object-oriented technology, specializing in C/C++, Java, Python, and Design Patterns ▪ Developed a multi-media training course in C programming ("Thinking in C") jointly with Mindview, Inc.

1990 - 1997: Consultant, The Church of Jesus Christ of Latter-day Saints, Salt Lake City UT
Development manager for the Church's first reusable, software infrastructure components project in a Windows NT client/server environment, using the Fusion object-oriented methodology, and Microsoft Visual C++ ▪ Conceived, designed, and implemented reusable, object-oriented frameworks for persistence and security for layered, object-oriented data models with relational databases ▪ Developed and championed software systems architecture ▪ Chairman of an in-house C Language Support Committee, which provided technical support, training, standards and direction for the use of C and C++ in software development ▪ Technical lead for a project that successfully automated the Church mission offices worldwide ▪ Designed and implemented a reporting subsystem in Oracle/ProC under HP-UX on-location for the Brazil Area Office

1984 - 1990: Member of the Technical Staff, Hughes Aircraft Company (now Raytheon), Tucson AZ
Developed software tools for data reduction and document preparation in C ▪ Developed a software defect tracking system in a relational DBMS (INGRES on VAX/VMS) ▪ Directed software configuration management for the TOW Missile Program ▪ Participated in the development of a rule-based missile diagnostic system in a Symbolics Common LISP environment ▪ Developed a C-language library for PC network communications based on NetBios ▪ VMS system manager for two MicroVAX systems networked with PCs via TCP/IP ▪ Managed a PC open shop for employee users ▪ Gave on-site training in C, relational databases, numerical methods and object-oriented programming

1981 - 1983: Research Assistant, University of Arizona, Tucson AZ
Performed scientific programming in FORTRAN and Ratfor under UNIX while a full-time Ph.D. candidate in Applied Mathematics (funded by a NASA Fellowship - finished all but dissertation with 4.0 gpa) ▪ Solved integral equations for water-flow research ▪ Wrote tools for automatic stability analysis of matrix algorithms

1978 - 1981: Member of the Technical Staff, Logicon, Inc., San Pedro CA
Developed software tools in FORTRAN in an IBM OS/MVS environment for the Space Defense Systems project ▪ Participated in the development of a software design/analysis tool written for IBM Corporation ▪ Developed software for weapons allocation and damage analysis studies ▪ Developed in-house training materials on APL and IBM's SPF software creation environment

Related Experience

- Columnist and Technical Editor for *Better Software Magazine*, 2007 – present
- Software consulting for Homeland Security with Mission Support, Inc. (2006-present)
- Developed and delivered Python training to software testers at Symantec, Inc. (2005, 2007)
- Editor of *The C/C++ Users Journal*, 1992-2003 (Senior Editor 2001-2003)
- Taught C++ and Java programming at various corporate sites for various training companies (Technology Exchange, Software Architects), 1993 – 1998
- Represented the Digital Equipment Corporation Users Society (DECUS), the Church of Jesus Christ of Latter-day Saints, and Utah Valley State College on the *ANSI/ISO Standards Committee* for the C++ Programming Language ▪ Designed part of the standard C++ library (1991-2001)
- Chair of the Industry Conference, *C/C++ Solutions*, R&D Publications, Kansas City, January 1995
- Developed and taught on-site courses in Numerical Methods, C Programming, Object-oriented Programming, and Relational Database theory and practice for Hughes Aircraft, 1987-1990

Publications

Books

1. Eckel, B. & Allison, C. (2004), *Thinking in C++, Volume 2: Practical Programming*, Prentice-Hall, ISBN 0130353132 (Czech Translation 2006, Chinese Translation 2008).
2. Allison, C. (1998), *C & C++ Code Capsules*, Prentice-Hall, ISBN 0135917859. (Chinese translation 2003).

Peer-Reviewed Papers

1. Allison, C. and Harrison, N. (2007 Oct) Teaching Design Patterns: A Matter of Principle, *The Journal of Computing Sciences in Colleges*, Volume 22, <u>10</u>, October 2007.
2. Allison, C. (2007 Oct) The Untapped Power of Generic Algorithms, *The Journal of Computing Sciences in Colleges*, Volume 22, <u>10</u>, October 2007.
3. Allison, C. (2007 Oct) Practical Computation Theory, *The Journal of Computing Sciences in Colleges*, Volume 22, <u>10</u>, October 2007.
4. Allison, C. (2007 Oct) The Simplest Automated Unit Test Tool That Could Possibly Work, *The Journal of Computing Sciences in Colleges*, Volume 22, <u>10</u>, October 2007.
5. Allison, C. (2005 Oct) Where Did All My Decimals Go?, *The Journal of Computing Sciences in Colleges*, Volume 20, <u>10</u>, October 2005.
6. Allison, C. (2000 Sep) The Simplest Automated Unit Test Framework That Could Possibly Work, *C/C++ Users Journal*, <u>18</u>, No. 9.
7. Cockburn, A. & Allison, C. (1998 Jun) Object-oriented Analysis & Design, Part 2, *C/C++ Users Journal*, <u>1</u>, No. 6.
8. Cockburn, A. & Allison, C. (1998 May) Object-oriented Analysis & Design, Part 1, *C/C++ Users Journal*, <u>1</u>, Nos. 5.
9. Allison, C. (1996 May) Object Persistence via Relational Databases, *C/C++ Users Journal*, <u>15</u>, No. 5. (Reprinted as a *Dr. Dobbs* select publication in 1997).
10. Allison, C. (1978) On Determining Functions of Matrices, *Pi Mu Epsilon Journal*, Fall 1978.

Professional Articles

1. Allison, C., (2008 May) A "D" in Programming, Part 2, *Better Software Magazine*, 10(4).
2. Allison, C., (2008 Apr) A "D" in Programming, Part 1, *Better Software Magazine*, 10(3).
3. Allison, C., (2008 Feb/Mar) Programming on the Other Side of Complexity, *Better Software Magazine*, 10(2).
4. Allison, C., (2008 Jan) Designing Generic Software, *Better Software Magazine*, 10(1).
5. Allison, C., (2007 Dec) Tools for Our Times, *Better Software Magazine*, 9(12).
6. Allison, C. (2007 Nov) Buddy, Can You Paradigm?, *Better Software Magazine*, 9(11).
7. Allison, C. (2007 Sep) Neglected Algorithms: Making Reuse a Reality with STL, *Better Software Magazine*, 9(9).
8. Allison, C. (2007 Aug) Piles of Sand Redux, *Better Software Magazine*, 9(8).
9. Allison, C. (2007 Jun) Piles of Sand, *Better Software Magazine*, 9(6).
10. Allison, C. (2007 Apr) The Roof is Going to Go, *Better Software Magazine*, 9(4).
11. Allison, C. (2007 Feb) Principle-Centered Software Development, *Better Software Magazine*, 9(2).
12. Allison, C. (2001 Aug) Reflection, Java Supplement to the *C/C++ Users Journal*, <u>19</u>, No. 8.
13. Allison, C. (2001 Jun) Threads 101, Java Supplement to the *C/C++ Users Journal*, <u>19</u>, No. 6.
14. Allison, C. (2001 Apr) Understanding Java Exceptions, Java Supplement to the *C/C++ Users Journal*, <u>19</u>, No. 4.
15. Allison, C. (2001 Feb) File Processing, Java Supplement to the *C/C++ Users Journal*, <u>19</u>, No. 2.
16. Allison, C. (2000 Nov) Basic Stream I/O, *C/C++ Users Journal*, <u>18</u>, No. 11.
17. Allison, C. (2000 Sep) Collections and Algorithms, *C/C++ Users Journal*, <u>18</u>, No. 9.
18. Allison, C. (2000 Jul) Locales and Formatted I/O, *C/C++ Users Journal*, <u>18</u>, No. 7.
19. Allison, C. (2000 Jun) Strings, *C/C++ Users Journal*, <u>18</u>, No. 6.
20. Allison, C. (2000 Mar) Arrays, *C/C++ Users Journal*, <u>18</u>, No. 3.
21. Allison, C. (2000 Jan) Interfaces and Inner Classes, *C/C++ Users Journal*, <u>18</u>, No. 1.
22. Allison, C. (1999 Nov) Object-oriented Programming in Java, *C/C++ Users Journal*, <u>17</u>, No. 11.
23. Allison, C. (1999 Sep) Packaging Your Objects, *C/C++ Users Journal*, <u>17</u>, No. 9.
24. Allison, C. (1999 Jul) Thinking in Objects, *C/C++ Users Journal*, <u>17</u>, No. 7.
25. Allison, C. (1999 May) Control Flow: The Bad, The Good, The Exceptional, *C/C++ Users Journal*, <u>17</u>, No. 5.

26. Allison, C. (1999 Mar) Using Primitive Types and Wrappers, *C/C++ Users Journal*, <u>17</u>, No. 3.
27. Allison, C. (1999 Jan) Jumping into Java, *C/C++ Users Journal*, <u>17</u>, No. 1.
28. Allison, C. (1998 Dec) What's New in Standard C++?, *C/C++ Users Journal*, <u>16</u>, No. 12.
29. Allison, C. (1997 Dec) Error Handling with C++ Exceptions, Part 2, *C/C++ Users Journal*, <u>15</u>, No. 12.
30. Allison, C. (1997 Nov) Error Handling with C++ Exceptions, Part 1, *C/C++ Users Journal*, <u>15</u>, No. 11.
31. Allison, C. (1996 Oct) C++: The Making of a Standard - Journey's End (An Interview with Bjarne Stroustrup), *C/C++ Users Journal*, <u>15</u>, No. 10.
32. Allison, C. (1995 May) Data Abstraction, *C/C++ Users Journal*, <u>14</u>, No. 5.
33. Allison, C. (1995 Apr) A Better C, *C/C++ Users Journal*, <u>14</u>, No. 4.
34. Allison, C. (1995 Mar) The Standard C Library, Part 3, *C/C++ Users Journal*, <u>14</u>, No. 3.
35. Allison, C. (1995 Feb) The Standard C Library, Part 2, *C/C++ Users Journal*, <u>14</u>, No. 2.
36. Allison, C. (1995 Jan) The Standard C Library, Part 1, *C/C++ Users Journal*, <u>14</u>, No. 1.
37. Allison, C. (1994 Dec) The Standard C++ Library, *C/C++ Users Journal*, <u>13</u>, No. 12.
38. Allison, C. (1994 Nov) Dynamic Memory Management, Part 2, *C/C++ Users Journal*, <u>13</u>, No. 11.
39. Allison, C. (1994 Oct) Dynamic Memory Management, Part 1, *C/C++ Users Journal*, <u>13</u>, No. 10.
40. Allison, C. (1994 Sep) Conversions and Casts, *C/C++ Users Journal*, <u>13</u>, No. 9.
41. Allison, C. (1994 Jul) C++ Exceptions, *C/C++ Users Journal*, <u>13</u>, No. 7.
42. Allison, C. (1994 Jun) Control Structures, *C/C++ Users Journal*, <u>13</u>, No. 6.
43. Allison, C. (1994 May) Visibility in C++, *C/C++ Users Journal*, <u>13</u>, No. 5.
44. Allison, C. (1994 Apr) Visibility in C, *C/C++ Users Journal*, <u>13</u>, No. 4.
45. Allison, C. (1994 Mar) The Preprocessor, *C/C++ Users Journal*, <u>13</u>, No. 3.
46. Allison, C. (1994 Feb) Variable-length Argument Lists, *C/C++ Users Journal*, <u>13</u>, No. 2.
47. Allison, C. (1994 Jan) Bit Handling in C++, Part 2, *C/C++ Users Journal*, <u>13</u>, No. 1.
48. Allison, C. (1993 Dec) Bit Handling in C++, Part 1, *C/C++ Users Journal*, <u>12</u>, No. 12.
49. Allison, C. (1993 Nov) Bit Handling in C, *C/C++ Users Journal*, <u>12</u>, No. 11.
50. Allison, C. (1993 Oct) Pointers, Part 3, *C/C++ Users Journal*, <u>12</u>, No. 10.
51. Allison, C. (1993 Sep) Pointers, Part 2, *C/C++ Users Journal*, <u>12</u>, No. 9.
52. Allison, C. (1993 Aug) Pointers, Part 1, *C/C++ Users Journal*, <u>12</u>, No. 8.
53. Allison, C. (1993 Jul) C++ Streams, *C/C++ Users Journal*, <u>12</u>, No. 7.
54. Allison, C. (1993 Jun) File Processing, Part 2, *C/C++ Users Journal*, <u>12</u>, No. 6.
55. Allison, C. (1993 May) File Processing, Part 1, *C/C++ Users Journal*, <u>12</u>, No. 5.
56. Allison, C. (1993 Apr) Sorting with qsort, *C/C++ Users Journal*, <u>12</u>, No. 4.
57. Allison, C. (1993 Mar) A C++ Date Class, Part 2, *C/C++ Users Journal*, <u>12</u>, No. 3.
58. Allison, C. (1993 Feb) A C++ Date Class, Part 1, *C/C++ Users Journal*, <u>12</u>, No. 2.
59. Allison, C. (1993 Jan) Time and Date Processing in C, *C/C++ Users Journal*, <u>12</u>, No. 1.
60. Allison, C. (1992 Dec) Text Processing, Part 3, *C/C++ Users Journal*, <u>11</u>, No. 12.
61. Allison, C. (1992 Nov) Text Processing, Part 2, *C/C++ Users Journal*, <u>11</u>, No. 11.
62. Allison, C. (1992 Oct) Text Processing, Part 1, *C/C++ Users Journal*, <u>11</u>, No. 10.

Editorials

1. Allison, C. (2008 Jul/Aug) Software: Use At Your Own Risk, *Better Software Magazine*, 10(5).
2. Allison, C. (2003 Nov) The Incremental Developer, *C/C++ Users Journal*, <u>21</u>, No. 11.
3. Allison, C. (2003 Oct) The Structure of Software Development Process Evolution, *C/C++ Users Journal*, <u>21</u>, No. 10.
4. Allison, C. (2003 Sep) Human Activities, *C/C++ Users Journal*, <u>21</u>, No. 9.
5. Allison, C. (2003 Aug) Programmer Dreams and Stranger Things, *C/C++ Users Journal*, <u>21</u>, No. 8.
6. Allison, C. (2003 Jul) A Matter of Trust, *C/C++ Users Journal*, <u>21</u>, No. 7.
7. Allison, C. (2003 Jun) Do Not Duplicate, *C/C++ Users Journal*, <u>21</u>, No. 6.
8. Allison, C. (2003 May) Got Quality?, *C/C++ Users Journal*, <u>21</u>, No. 5.
9. Allison, C. (2003 Apr) Stupid Tricks, *C/C++ Users Journal*, <u>21</u>, No. 4.
10. Allison, C. (2003 Mar) The Proactive Programmer, *C/C++ Users Journal*, <u>21</u>, No. 3.

11. Allison, C. (2003 Feb) The C++ Experience, *C/C++ Users Journal*, <u>21</u>, No. 2.
12. Allison, C. (2003 Jan) The Wit to Win, *C/C++ Users Journal*, <u>21</u>, No. 1.
13. Allison, C. (2002 Dec) Roots II, *C/C++ Users Journal*, *20*, No. 12.
14. Allison, C. (2002 Nov) Part of the Landscape, *C/C++ Users Journal*, <u>20</u>, No. 11.
15. Allison, C. (2002 Oct) Euclid Alone, *C/C++ Users Journal*, <u>20</u>, No. 10.
16. Allison, C. (2002 Sep) The Best of Both?, *C/C++ Users Journal*, <u>20</u>, No. 9.
17. Allison, C. (2002 Aug) New Cheese, *C/C++ Users Journal*, <u>20</u>, No. 8.
18. Allison, C. (2002 Jul) O Brother, Where Art Thou?, *C/C++ Users Journal*, <u>20</u>, No. 7.
19. Allison, C. (2002 Jun) Gear-head Humor and Other Important Items, *C/C++ Users Journal*, <u>20</u>, No. 6.
20. Allison, C. (2002 May) Is Portability Still Important?, *C/C++ Users Journal*, <u>20</u>, No. 5.
21. Allison, C. (2002 Apr) Good Stuff, *C/C++ Users Journal*, <u>20</u>, No. 4.
22. Allison, C. (2002 Mar) SMOP, *C/C++ Users Journal*, <u>20</u>, No. 3.
23. Allison, C. (2002 Feb) In-betweeners, *C/C++ Users Journal*, <u>20</u>, No. 2.
24. Allison, C. (2002 Jan) A Simple Collection of Bits, *C/C++ Users Journal*, <u>20</u>, No. 1.
25. Allison, C. (2001 Dec) What Every Programmer Should Know, *C/C++ Users Journal*, <u>19</u>, No. 12.
26. Allison, C. (2001 Nov) We Have a Winner, *C/C++ Users Journal*, <u>19</u>, No. 11.
27. Allison, C. (2001 Oct) Do Your Homework, *C/C++ Users Journal*, <u>19</u>, No. 10.
28. Allison, C. (2001 Sep) This Sentence is False, *C/C++ Users Journal*, <u>19</u>, No. 9.
29. Allison, C. (2001 Aug) The Name Game, *C/C++ Users Journal*, <u>19</u>, No. 8.
30. Allison, C. (2001 Jul) Something Cool, *C/C++ Users Journal*, <u>19</u>, No. 7.

Lectures

Keynote Lectures

1. "Tools for Our Times: Tales of Power, Intrigue and Serendipity in Computing", Keynote Address at the Intermountain conference of the *Consortium for Computing Sciences in Colleges* (CCSC), October 2004.
2. "Practical Excellence: A Perspective on Code Quality," Keynote Address, *Association of C/C++ Users* (Oxford, UK), April 2004.

Invited Lectures

1. "The Design Principles Behind Design Patterns", *Better Software Conference*, June 2008.
2. "Functional Programming Makes a Comeback", *Better Software Conference*, June 2008.
3. "Fundamentals of Floating-point Arithmetic", *In-Service Workshop for Teachers of Science and Engineering*, UVSC, June 2008.
4. "Totally Awesome Computing: Python as a General-Purpose Object-Oriented Programming Language", *ACM/OOPSLA*, 2006, 2007.
5. "Understanding C++ Templates", *Software Development Conference*, March 2005 – 2008, October 2006, 2007.
6. "Where Did All My Decimals Go?", *Software Development Conference*, March 2006.
7. "Got Quality?", *Software Development Conference*, March 2005, 2006.
8. "Understanding C++ Templates," *Association of C/C++ Users* (Oxford, UK), April 2004.
9. "Automated Testing", *Provo Linux Users Group*, Orem, UT, January 2003.
10. "The Forgotten Containers", *Software Development Conference*, 2002.
11. "Understanding C++ Exceptions", *Software Development Conference*, 1997 - 2003.
12. "Java Threads", *Software Development Conference*, 2001-2003.
13. "The Simplest Automated Unit Test Framework That Could Possibly Work", *Software Testing, and Review Conference (STAR)*, 2001; also *Software Development Conference*, 2001.
14. "The Java 2 Collections", *Software Development Conference*, 1999-2001.
15. "Leveraging the Standard C++ Library", *Software Development Conference*, 1994-1999.
16. Gave several invited lectures on C++ programming at *DECUS Symposia*, 1991-1993.

Professional Development

- Attended the *International Conference on Functional Programming*, September 2008
- Attended the first annual D Developers Conference, Seattle, WA, August 2007.
- Attended various educational talks and tutorials at *Software Development Conference* twice yearly, 1994 – 2008
- Attended *C++ Connections*, Las Vegas, Nevada, November 2005
- Attended the regional conference of *The Consortium for Computing Sciences in Colleges* (CCSC), 2004, 2005, 2007
- Attended *PyCon*, the Python Community Annual Conference, Washington, D.C., March 14-18, 2005
- Attended the annual Symposium of *SIGCSE*, the Special Interest Group for Computer Science Education, Reno, Nevada, February 2003
- Attended *Java One*, San Francisco, California, June 2004
- Attended the *Agile Development Conference*, Salt Lake City, Utah, June 2003
- Co-Convener of "Summit on Code Quality", Portland, Oregon, January 2003
- Attended a seminar on Advanced Java Programming, *Mindview, Inc.*, Crested Butte, Colorado, July 1998
- Attended *DECUS Symposia* semi-annually, 1991-1993.