CSTA K12 Standards in Action in Your Classroom

Michelle Lagos, Karen Lang, Tammy Pirmann, Deborah Seehorn

CSTA Annual Conference
July 16, 2013
Agenda

• Attendee Introductions
• Overview of CSTA K12 Standards
  • Handout Standards Booklets
• Crosswalk Documents
  • Handout Templates
  • http://csta.acm.org/Curriculum/sub/K12Standards.html
• Example Crosswalks
• Hands-on Crosswalk Document Work
  • break into groups by school/levels???
CSTA K12 Standards
Knowledge for Today and Beyond

We consider it critical that students be able to read and write and understand the fundamentals of math, biology, chemistry and physics. To be a well-educated citizen in today’s computing-intensive world, students must have a deeper understanding of the fundamentals of computing as well.

New Standards Task Force

Deborah Seehorn
   North Carolina Department of Public Instruction, Chair
Stephen Carey
   Brunswick School Department
Daniel Moix
   Ouachita Technical College
Barb Owens
   Southwestern University

Brian Fuschetto
   Lyndhurst High School
Irene Lee
   Santa Fe Institute
Dianne O’Grady-Cuniff
   Howard High School
Chris Stephenson
   Computer Science Teachers Association
Anita Verno
   Bergen Community College
Why Standards?

• Many states have a computer education requirement at the K-12 grade level but this has many different meanings.
• General computer knowledge and skills have been moving…
  • Traditional HS courses may now be in elementary and middle school
  • Keyboarding, General Computers, Office Programs, Computing Concepts are all clumped under "computing courses".
• Trends in the High School Curriculum
  • CS is found in an elective environment
  • Focus is on Standards and Assessment
  • Computer Teachers – Certification requirements vary (if existent!)
The ACM/CSTA Model Curriculum

- Published in:
  - 2003
  - 2006: revised forward
- More than 40,000 copies distributed
- Used as the basis for curriculum development at the state and national level
Context for New Standards

• CSTA Model Curriculum was last revised in 2006

• Much has been learned since then, including how to write standards that are consistent in format with those of other disciplines

• New tools and pedagogies have been developed to make computer science more accessible for all students

• There is still confusion between educational technology (the use of computers to support learning in other disciplines) and computer science

• We define computer science as:

  “Computer science (CS) is the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society.”
CSTA K–12 Computer Science Standards

Revised 2011

The CSTA Standards Task Force
Getting the Standards

You can access a downloadable, hyperlinked version of the standards at:

http://csta.acm.org/Curriculum/sub/K12Standards.html

To request hardcopies for advocacy events, contact:

Chris Stephenson  
cstephenson@csta.acm.org
Standards are Organized into Levels

Level 1
Grades K-6
- Computer Science and Me

Level 2
Grades 6-9
- Computer Science and Community

Level 3
Grades 9-12
- Computer Science in the Modern World
  - Computer Science Principles
  - Topics in Computer Science

Applying Concepts and Creating Real-World Solutions
Level Definitions

- **Level 1** (recommended for grades K–6)
  *Computer Science and Me*

- **Level 2** (recommended for grades 6–9)
  *Computer Science and Community*

- **Level 3** (recommended for grades 9–12)
  *Applying concepts and creating real-world solutions*
Level Definitions

• **Level 1** (recommended for grades K–6)  
  *Computer Science and Me*

• **Level 2** (recommended for grades 6–9)  
  *Computer Science and Community*

• **Level 3** (recommended for grades 9–12)  
  *Applying concepts and creating real-world solutions*
Level Definitions

- **Level 1** (recommended for grades K–6)  
  *Computer Science and Me*

- **Level 2** (recommended for grades 6–9)  
  *Computer Science and Community*

- **Level 3** (recommended for grades 9–12)  
  *Applying concepts and creating real-world solutions*
Level 3 Course Descriptions

• **Level 3A**: (recommended for grades 9 or 10) *Computer Science in the Modern World*

• **Level 3B**: (recommended for grades 10 or 11) *Computer Science Concepts and Practices*

• **Level 3C**: (recommended for grades 11 or 12) *Topics in Computer Science*
Level 3 Course Descriptions

• Level 3A: (recommended for grades 9 or 10)
  *Computer Science in the Modern World*

• Level 3B: (recommended for grades 10 or 11)
  *Computer Science Concepts and Practices*

• Level 3C: (recommended for grades 11 or 12)
  *Topics in Computer Science:*
Level 3 Course Descriptions

• **Level 3A:** (recommended for grades 9 or 10) *Computer Science in the Modern World*

• **Level 3B:** (recommended for grades 10 or 11) *Computer Science Concepts and Practices*

• **Level 3C:** (recommended for grades 11 or 12) *Topics in Computer Science*
Learning Outcomes Organized by Strands

- Computational Thinking
- Collaboration
- Community, Global, and Ethical Impacts
- Computers and Communication Devices
- Computing Practice
Example Strand for Level 2

Computing Practice & Programming / Careers

The student will be able to:

1. Use a variety of multimedia tools and peripherals to support personal productivity and learning throughout the curriculum.
2. Select appropriate tools and technology resources to accomplish a variety of tasks and solve problems.
3. Design, develop, publish, and present products (e.g., webpages, mobile applications, animations) using technology resources that demonstrate and communicate curriculum concepts.
4. Implement problem solutions using a programming language, including: looping behavior, conditional statements, logic, expressions, variables, and functions.
5. Demonstrate an understanding of algorithms and their practical application.
6. Demonstrate good practices in personal information security using passwords, encryption, and secure transactions.
7. Demonstrate dispositions amenable to open-ended problem solving and programming (e.g., comfort with complexity, persistence, brainstorming, adaptability, patience, propensity to tinker, creativity, accepting challenge).
8. Identify interdisciplinary careers that are enhanced by computer science.
9. Collect and analyze data that is output from multiple runs of a computer program.
Computing Practice & Programming Strand

• Using technology resources for learning
• Using technology tools for the creation of digital artifacts
• Programming
• Interacting with Remote Information
• Careers
• Data Collection and Analysis
Computing Practice Scaffolding Map

Level 3B
- Use object-oriented principles in program design
- Explore principles of system design in scaling, efficiency, and security

Level 3A
- Use advanced tools to create digital artifacts
- Use functions and classes to decompose large-scale computational problems
- Classify programming languages by level and application domain
- Use APIs and libraries to facilitate programming solutions
- Describe the variety of programming languages available
- Solve problems using analysis, design and implementation techniques
- Explain the program execution process
- Select appropriate file systems and data structures

Level 3
- Deploy principles of security by examining encryption and authentication techniques
- Anticipate future careers and the technologies that will exist
- Use data analysis to enhance understanding of complex natural and human systems
- Compare techniques for analyzing massive data collections
- Explore a variety of careers in computing IT specialist, web designer, systems analyst
- Describe techniques for locating

Level 2
- Use a variety of multimedia tools and programs to select personal productivity and learning strategies
- Design, develop, publish and present products using technology resources that demonstrate and communicate curriculum concepts
- Implement problem solutions in a programming environment using looping, branch decision, conditional statements, input, output, variables, and functions
- Demonstrate an understanding of algorithms, and their practical applications
- Develop dispositions amenable to open-ended problem solving and programming
- Identify interdisciplinary careers that are enhanced by computer science
- Collect and analyze data that is output from multimedia tools of a computer program

Level 1B
- Use technology resources to address a variety of tasks and problems
- Use technology tools for individual and collaborative writing, communication, and publishing activities
- Construct a program as a set of statements to be acted out
- Use computing devices, including mobile devices, to access remote information, communicate with others in support of direct and independent learning and pursue personal interests
- Identify different careers in computing and the interdisciplinary nature of computing in the 21st century
- Gather and organize information using a variety of digital concept mapping tools

Level 1A
- Use technology resources to conduct age-appropriate research
- Create developmentally appropriate multimedia products with support from teachers, family members, or student partners
- Construct a set of statements to be acted out
- Identify careers that are wide spectrum of software engineering and technology
- Gather and organize information with concept mapping tools

Level 1
- Use technology resources for learning
- Use technology tools for the creation of digital artifacts
- Use technology tools for the creation of digital artifacts
- Programming

CSTA
Computer Science Teachers Association
Computing Practice Scaffolding Map

Level 3B

- Use object-oriented principles in program design
- Use advanced tools to create digital artifacts
- Use mobile devices to design, develop, and implement mobile computing applications
- Use digital tools to create digital artifacts
- Use technologies to facilitate programming solutions
- Solve problems using analysis, design, and implementation techniques
- Use debugging and unit testing to verify programs
- Describe the variety of programming languages available
- Select appropriate file formats for various types of data
- Classify programming languages by level and application domain
- Explore the principles of system design in scaling, efficiency, and security
- Deploy principles of security by examining encryption and authentication techniques
- Anticipate future directions for technologies that facilitate
- Deploy various data collection techniques for different types of problems
- Use data analysis to enhance understanding of complex natural and human systems
- Compare techniques for analyzing massive data collections
- Describe techniques for locating and collecting small and large scale data sets

Level 3A

- Explain the principles of security by analyzing encryption, cryptography, and authentication techniques
- Explore a variety of career opportunities involving computing IT specialists, systems analysts, software programmers, CIOs, computer scientists
- Develop dispositions toward open-ended problem solving
- Demonstrate the professional behavior expected of a computing IT specialist, systems analyst, software programmer
- Identify and apply cognitive skills that are enhanced by computer applications
- Understand the organization of internal elements and web pages
- Use computing devices, including mobile devices, to access remote information, communicate with others in support of direct and independent learning, and pursue personal interests
- Identify different careers involving computing and the important aspects of computing in the 21st century
- Gather and organize information using a variety of digital concept mapping tools
- Identify careers in computer technology
- Gather and organize information with concept mapping tools

Level 2

- Design, develop, publish, and present products using technology resources that demonstrate and communicate curriculum content
- Demonstrate an understanding of algorithms, and their practical applications
- Implement problem solutions in a programming environment using loops, functions, data structures, and control statements
- Develop dispositions towards open-ended problem solving
- Demonstrate good practices in personal information security, using passwords, encryption, and data transactions
- Identify interpersonal skills that are enhanced by computer applications
- Understand the organization of internal elements and web pages
- Use computing devices, including mobile devices, to access remote information, communicate with others in support of direct and independent learning, and pursue personal interests
- Identify different careers involving computing and the important aspects of computing in the 21st century
- Gather and organize information using a variety of digital concept mapping tools

Level 1B

- Use technology resources for individual and collaborative writing, research, and publishing activities
- Construct a program as a set of statements to be acted out
- Implement problem solutions in a block-based visual programming environment
- Use computing devices, including mobile devices, to access remote information, communicate with others in support of direct and independent learning, and pursue personal interests
- Identify different careers involving computing and the important aspects of computing in the 21st century
- Gather and organize information using a variety of digital concept mapping tools

Level 1A

- Use technology resources to conduct age-appropriate research
- Use technology tools for the creation of digital artifacts
- Use technology resources for learning
- Create developmentally-appropriate multimedia products with support from teachers, family members, or student partners
- Construct a set of statements to be acted out
- Interacting with remote information
- Identify careers in computer technology
Looking at a learning progression

For example, in the Computational Thinking strand there is a column on Modeling and simulation.

How do learning experiences at each level build upon the previous level?
How Can the Standards Be Used?

• Curriculum Design
• Curriculum Review / Modification
• Improving Instruction
• Advocacy
• Planning Resource
• Reference
• Teacher Certification
Why are K-12 CS Standards So Important?

Atlantic
June 21, 2012

How America Can Get More Start-Up Talent

Build it right at home: Congress should encourage public schools to teach American children how to code just after they learn to multiply.

In an op-ed in US News and World Report (6/22), inventor and entrepreneur Dean Kamen writes, "What we need in this country is a return to a culture that celebrates new discoveries and brainpower, a culture that embraces the challenges and rewards of innovation. Imagine a nation of curious inventors and tinkerers finding a cure for cancer, or creating energy solutions for the 21st century." Kamen writes, "Google started with a spark of genius by two entrepreneurs and mushroomed into a tech giant that provides instant free access to information, not to mention global science fairs that encourage kids to explore their creativity." However, "we need more from businesses-resources, innovative thinking, and employee volunteer talent. By supporting STEM initiatives from the White House and the US Department of Education, public/private partnerships have the opportunity to train and excite young talent and turn them into technology leaders."
Job Outlook for Software Developers

Bureau of Labor Statistics
1. Software Engineer

2012 pay: $88,142
2011 pay: $87,140
Change in pay: +1.1%

Software engineer has been the No. 1 job for two years running. You can sum that up in two words, Lee said: “Technology revolution.”

Software engineers are the “creative minds behind computer programs,” according to the

Read More

Photo: Bloomberg via Getty Images
Crosswalk Documents

http://csta.acm.org/Curriculum/sub/K12Standards.html
# CS Standards Crosswalk with CSTA K-12 Computer Science Standards for State/District/Course Standards

http://csta.acm.org/Curriculum/sub/K12Standards.html

<table>
<thead>
<tr>
<th>CS Standards Name</th>
<th>Website</th>
<th>Contact Info</th>
</tr>
</thead>
</table>

Please indicate whether your standards are: State □ District □ School □ or Institution □

<table>
<thead>
<tr>
<th>CSTA Standard Level/Grade Level</th>
<th>Strand</th>
<th>CS Standard</th>
<th>Aligned Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1 (recommended for grades K–6) <strong>Computer Science and Me</strong> Elementary school students are introduced to foundational concepts in computer science by integrating basic skills in technology with simple ideas about computational thinking. The learning experiences created from these standards should be inspiring and engaging, helping students see computing as an important part of their world.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 1/ K-3</td>
<td>Computational Thinking</td>
<td>Use technology resources (e.g., puzzles, logical thinking programs) to solve age-appropriate problems.</td>
<td></td>
</tr>
<tr>
<td>Level 1/ K-3</td>
<td>Computational Thinking</td>
<td>Use writing tools, digital cameras, and drawing tools to illustrate thoughts, ideas, and stories in a step-by-step manner.</td>
<td></td>
</tr>
<tr>
<td>Level 1/ K-3</td>
<td>Computational Thinking</td>
<td>Understand how to arrange (sort) information into useful order, such as sorting students by birth date, without using a computer.</td>
<td></td>
</tr>
<tr>
<td>Level 1/ K-3</td>
<td>Computational Thinking</td>
<td>Recognize that software is created to control computer operations.</td>
<td></td>
</tr>
<tr>
<td>Level 1/ K-3</td>
<td>Computational Thinking</td>
<td>Demonstrate how 0s and 1s can be used to represent information.</td>
<td></td>
</tr>
<tr>
<td>CSTA Standard Level/Grade Level</td>
<td>Strand</td>
<td>CS Standard</td>
<td>Aligned Standard</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>--------</td>
<td>-------------</td>
<td>------------------</td>
</tr>
<tr>
<td><strong>Level 1 (recommended for grades K–6) Computer Science and Me</strong> Elementary school students are introduced to foundational concepts in computer science by integrating basic skills in technology with simple ideas about computational thinking. The learning experiences created from these standards should be inspiring and engaging, helping students see computing as an important part of their world.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Use general-purpose productivity tools and peripherals to support personal productivity, remediate skill deficits, and facilitate learning.</td>
<td>• Use proven typing program <em>Grade 6, Topic 4, Typing Skills</em></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Use technology tools (e.g., multimedia and text authoring, presentation, web tools, digital cameras, and scanners) for individual and collaborative writing, communication, and publishing activities.</td>
<td>• Use Paint and web tools to build and create animation <em>Grade 6, Topic 6, Elementary Computer Graphics</em></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Gather and manipulate data using a variety of digital tools.</td>
<td></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Construct a program as a set of step-by-step instructions to be acted out (e.g., make peanut butter and jelly sandwich activity).</td>
<td>• Notion of algorithms <em>Grade 6, Topic 13, Designing Solutions, Pseudo Code, Flowchart, Coding Algorithms</em></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Implement problem solutions using a block based visual programming language.</td>
<td>• Using Alice visual programming  • Programming as steps to solve a problem  • Concept of an object, methods, looping <em>Grade 6, Topic 14, Intro to Programming</em></td>
</tr>
<tr>
<td><strong>Level 1/3-6</strong></td>
<td>Computing Practice and Programming</td>
<td>Use computing devices to access remote information, communicate with others in support of direct and independent learning, and pursue</td>
<td></td>
</tr>
</tbody>
</table>
## Curriculum Resource Crosswalk with CSTA K-12 Computer Science Standards for Resources/Texts

[http://csta.acm.org/Curriculum/sub/K12Standards.html](http://csta.acm.org/Curriculum/sub/K12Standards.html)

<table>
<thead>
<tr>
<th>Curriculum Provider Website/Contact Info</th>
<th>Name of Resource</th>
<th>Type of Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CSTA Standard Level/Grade Level</th>
<th>Strand</th>
<th>CS Standard</th>
<th>Location of Resource (Page number and/or URL)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Level 1</strong> (recommended for grades K–6) <strong>Computer Science and Me</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elementary school students are introduced to foundational concepts in computer science by integrating basic skills in technology with simple ideas about computational thinking. The learning experiences created from these standards should be inspiring and engaging, helping students see computing as an important part of their world.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Level 1</strong> K-3</td>
<td>Computational Thinking</td>
<td>Use technology resources (e.g., puzzles, logical thinking programs) to solve age-appropriate problems.</td>
<td></td>
</tr>
<tr>
<td><strong>Level 1</strong> K-3</td>
<td>Computational Thinking</td>
<td>Use writing tools, digital cameras, and drawing tools to illustrate thoughts, ideas, and stories in a step-by-step manner.</td>
<td></td>
</tr>
<tr>
<td><strong>Level 1</strong> K-3</td>
<td>Computational Thinking</td>
<td>Understand how to arrange (sort) information into useful order, such as sorting students by birth date, without using a computer.</td>
<td></td>
</tr>
<tr>
<td><strong>Level 1</strong> K-3</td>
<td>Computational Thinking</td>
<td>Recognize that software is created to control computer operations.</td>
<td></td>
</tr>
<tr>
<td><strong>Level 1</strong> K-3</td>
<td>Computational Thinking</td>
<td>Demonstrate how 0s and 1s can be used to represent information.</td>
<td></td>
</tr>
<tr>
<td>CSTA Standard Level/Grade Level</td>
<td>Strand</td>
<td>CS Standard</td>
<td>Location of Resource (Page number and/or URL)</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computing Practice and</td>
<td>Explain the principles of security by examining encryption, cryptography, and authentication techniques.</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computing Practice and</td>
<td>Explore a variety of careers to which computing is central.</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computing Practice and</td>
<td>Describe techniques for locating and collecting small and large-scale data sets.</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computing Practice and</td>
<td>Describe how mathematical and statistical functions, sets, and logic are used in computation.</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computers and</td>
<td>Describe the unique features of computers embedded in mobile devices and vehicles (e.g., cell phones, automobiles, airplanes).</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computers and</td>
<td>Develop criteria for purchasing or upgrading computer system hardware.</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computers and</td>
<td>Describe the principal components of computer organization (e.g., input, output, processing, and storage).</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computers and</td>
<td>Explain the multiple levels of hardware and software that support program execution (e.g., compilers, interpreters, operating systems, networks).</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3A/9-12</td>
<td>Computers and</td>
<td>Explain the basic components of computer networks (e.g., servers, file protection, routing, spoolers and queues, shared resources, and fault-tolerance).</td>
<td><a href="http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials">http://www.lpi.org/linux-certifications/introductory-programs/linux-essentials</a></td>
</tr>
<tr>
<td></td>
<td>Communication Devices</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Hands-on

Small group work on individual crosswalks
Presenters' Contact Information

- Deborah Seehorn, CSTA Curriculum Committee Chair
  deborah.seehorn@dpi.nc.gov
- Karen Lang
  klang@wpi.edu
- Michelle Lagos
  mlagos@amschool.org
- Tammy Pirmann
  tpirmann@gmail.com