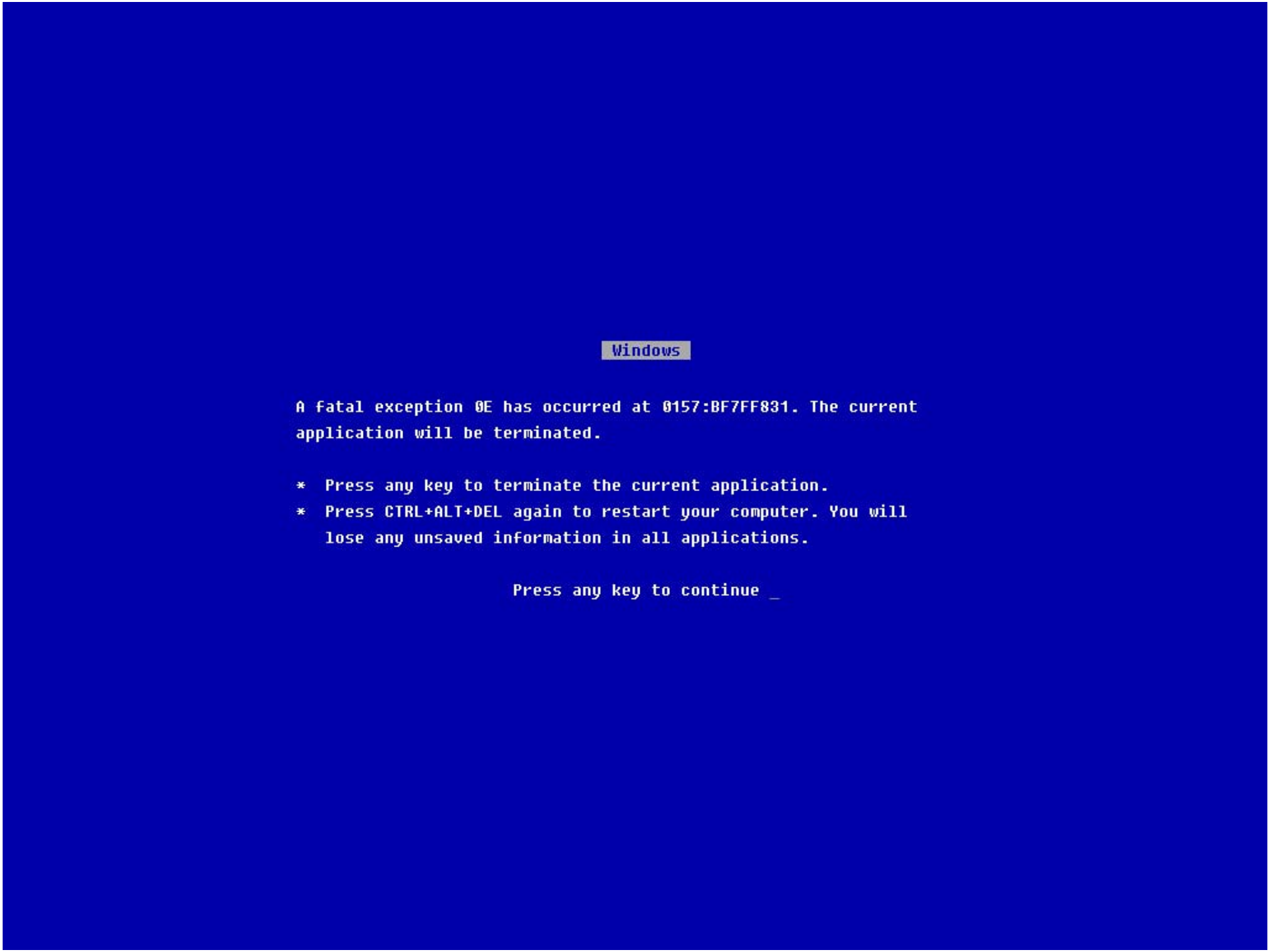
Computer Science Research and Writing

Gary T. Leavens
Com Sci., Iowa State Univ.
www.cs.iastate.edu/~leavens/
August 17, 2006

Why does this matter?

- Society depends on us


Windows
A fatal exception 0E has occurred at 0157:BF7FF831. The current application will be terminated.
* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will lose any unsaved information in all applications.
Press any key to continue _

Why does this matter?

- Society depends on us
- Your career and satisfaction depend on it
- Your ideas thrive only if others read them
- It’s challenging and fun

Main Point: Honesty

- Keep yourself honest
  - to find truth more rapidly
  - to avoid embarrassment
- Don’t oversell
  - present the problem
  - present the evidence
  - present the limits and assumptions

Outline

- Introduction
- Problem: making an impact
- Paradigms
  - theoretical, experimental
  - some advice for each kind
- Writing
  - talks, conference papers, journal papers
- Summary

Goal: Positive Impact

- Ideas, papers
- Systems

How Research is Judged

- Impact
  - Is it widely used?
  - Does it lead to new directions?
  - Does it affect teaching?
- Standards differ among venues
  - conferences: timely, interesting, simple, short
  - journals: correct, relevant, well-written

Fundamental Problems

- Lots of prior work
- Lots of researchers

Research Paradigms

- Theoretical
  - "Publish or perish!"
  - E.g., algorithms that solve real problems
  - Evaluation by proof, elegance, clarity
- Experimental or Systems
  - "Demo or die!"
  - Evaluation by experiment, simplicity, utility
- Also: Cross-Disciplinary, …

Theoretical Research

- Keep an eye on applications
  - great source of problems and interest
  - invest in an area that is starting to develop
- Look for “something to push against”
  - theorems relate two things
- Remember the costs
  - speed, space, complexity, etc.
- Strive for simplicity, elegance, clarity

Advice for Theoretical Research

- Stay “light on your feet”
  - Seek new approaches or simplifications
  - Don’t work on the same area forever
  - Have short-term goals
- Learn from writing
- Read
  - selectively and critically

Experimental / Systems Research

- Find ways to see farther (new data)
- Keep an eye on theory
  - validation or invalidation are both good
- Keep other eye on end-users
  - main source of problems, feedback
- Look for “something to push against”
  - the way to evaluate your system or demo
- Look for insights (lessons, theories)

Advice for Experimental / Systems

- Do separate, short projects
- Pick simple solutions, avoid the complex
- Seek feedback and evaluation
- Be sure to finish your project
- Do quantitative evaluation
- Do technology transfer
  - points above are from Dave Patterson

Cross-Disciplinary Research

- Apply computing to other disciplines
- Use computation for theory construction
- Evaluation in area of application:
  - Originality
  - Utility and results
- Evaluation in computing
  - Soundness and currency of the CS applied
  - Finding new CS problems

Ways to Make a Positive Impact

- Publish important work first
  - Think hard
  - Use new techniques/instruments
  - Work in underdeveloped area
  - Start new (sub-)area
- Publish clear descriptions
  - Relate to current understanding
- Be persistent

Finding Good Ideas

- Look for problems
  - In reading, teaching
  - By using your own tools / systems
- Have lots of ideas
- Pursue ones that:
  - You are uniquely qualified to handle
  - Tackle important problem
  - Excite you
  - You make progress on

How to Improve?

- Ask a lot of questions
  - "Why?"
- Read a lot
- Develop judgment about
  - Problems
  - Solution techniques
  - Explanations, evaluations

Becoming a Researcher

- Read widely and deeply
- develop judgement about great papers
- build general knowledge
- look for issues and questions
- capture opportunities
- keep a research notebook
- follow references and use the *Sci. Cit. Ind.*

Writing: Why does it Matter?

Determines if your ideas are:

- Published,
- Read,
- Understood,
- Remembered,
- Cited,
- Taught.

Who are they?

Who are these two computer scientists?

David L. Parnas

Barbara H. Liskov

Their Papers on Modular Design

David L. Parnas.
On the criteria to be used in decomposing systems into modules.
Commun. ACM, 15(12):1053.1058, Dec. 1972.

Barbara H. Liskov,
A Design Methodology for Reliable Software Systems,
1972 Fall Joint Computer Conference, pp. 191-199, AFIPS, 1972.

Writing

- The scientific style
- Advice
  - writing well
  - giving talks

The Scientific Style

- Purpose:
  - allow reader to judge the research
  - describe and present evidence
  - convey ideas and insights clearly
- Not:
  - impress the reader
  - make an artistic statement

Different Kinds of Writing

- Teaching (as in textbooks)
  - focus on explanation of science
  - breadth and clarity are most important
  - newness is not important
- Literature, poetry, etc.

Learning about Writing

- Read to observe the style
  - Journals in your field
  - *Scientific American*
  - Steven Jay Gould and other science writers
  - Storytellers: Mark Twain
- Observe how they
  - organize
  - explain

The Writing Process

- Start by “brainstorming”
- Organize the ideas (outline)
  - don’t “dump core”
- Once ideas on paper, make them clear
- Edit from a hard copy sometimes
- Seek feedback
- Enhance awareness by tracking problems
- Writing is rewriting

Writing my Dissertation

- Guttag's advice: "Keep a list"
- Hardest lesson: "Don't core dump"
- Writing is like programming

Writing ~ Programming (Theory)

Written Text	**Program**
Definition	Declaration
Theorem statement	Procedure interface (specification)
Proof	Implementation
Lemma	Subroutine
Remark	Comment
Example	Test case

Writing ~ Programming (Systems)

Written Text	**Program**
Definition	Declaration
Goal (or problem)	Procedure specification
Description of Code	Implementation
Subproblem	Subroutine
Application/Example	Comment
Performance results	Test case

Why the Analogy is Helpful

- Is it well organized?
- Is everything in the proper place?
- Is it maintainable?
- Is there repetition?
- Does it work?

Writing Related Work

- Related to *problem*
  - Not just to your solution technique
- Help reader fit your work into problem space

- Say how helps solve problem
- Say why / how doesn't solve problem
  - Also how solution techniques differ

Getting All the Related Work

- Read other dissertations
- Ask the experts
- Read the references in good papers
  - Science Citation Index
  - Recent conferences / journal issues
- You may need to go to the library!
- Peters and jmlunit story

How to Link Sentences

- Gopen & Swan, “Science of Scientific Writing” (*American Scientist*, 78:550-558, 1990)
- See *Style* by Williams (U. Chicago, 1990)

Non-linking

Sentences have 2 parts in English.

Links to previous material appear in **the first part**.

Emphasis and new information are provided by **the second part**.

Linking idea

In English

sentences have 2 parts.

The first part

links to previous material.

The second part

provides new information and emphasis.

Other Writing Ideas

- Illustrate with examples
  - Also counterexamples!
  - Especially anything initially unclear
- “Pair writing” with a professor
- Honesty
  - Present facts, don’t sell
  - Look for flaws

Lower-Level Tips

- Use signposting:
  - "This section describes the algorithm for …"
- Use topic sentences
  - "The key idea is to use a divide-and-conquer strategy."
- Don't use (very many) adjectives

Talks

- Standard outline for a technical talk:
  - problem and its importance
  - background
  - details of problem, solution
  - related work, future work
  - summary
- Use this outline recursively within the talk

Advice for Talks

- Don’t use too many words in making your point, just put up a few focus words on the slide, so that your audience won’t be distracted reading; you can always say all this anyway.
- Use pictures and graphs when possible
- Practice
- Ideas count, not the performance

Summary

- Look for “something to push against”
- Read selectively and critically
- Writing is like programming
- Strive for clarity
- Strive for honesty: don’t oversell

References

- G. D. Gopen and J. A. Swan, The Science of Scientific Writing, *American Scientist*, **78**:550-558, Nov-Dec. 1990.
- D. E. Knuth, et al., *Mathematical Writing*, MAA Notes, vol. 14, Math. Assoc. of America, 1989.
- http://www.goanna.cs.rmit.edu.au/~jz/writing.html