Perspective on Computer Science Education

Amiram Yehudai, Tel Aviv University

Issues

- Research and education
- CS college education, vs. CS K-12 education
  - what to teach
  - how to teach
- The role of programming Languages

Personal perspective

Personal History

- Main occupation: CS Faculty member
  - Research
  - Teaching
  - Administrative duties

- College level curriculum development in TAU
  - Models of Computation
  - Formal Languages
  - Compilers
  - Software Workshop

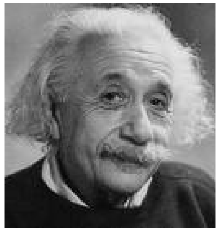
Research and teaching

- PhD thesis – Formal Language Theory
- soon after moved to Programming Languages and Software Engineering
- Influenced by a Consulting job
  - Structured programming in Assembly language
  - Using Automata to express design
- Interaction with ideas about teaching
  - Software Workshop

Programming

- Introduction to CS, using Pascal
  - What is the essence of programming?
  - What is the best way to teach it?
  - In what order to teach?
    - Teach X as early as possible, but not earlier.
    - Similar question today – Objects first?
    - No "one size fits all" for students/teachers
- Text books

Algorithms + Data Structures = Programs

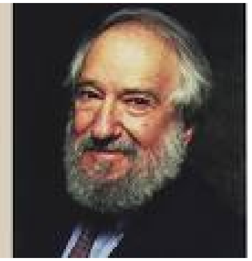


Programming Languages

- The Ada language
  - Doing vs. Explaining
- Influence of programming Languages
- Lisp (later Scheme)
- Setl : executable specification
- Prolog : prototyping
- Eiffel: Design by Contract
- Introducing a new subject

More on Programming Languages

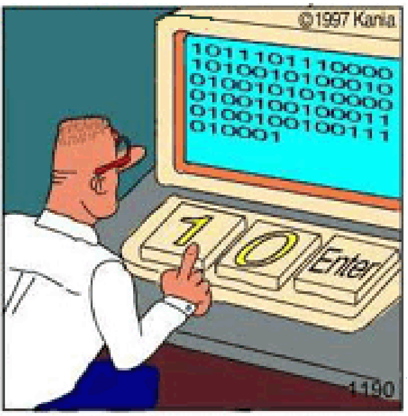
- “A programming language is like a natural, human language in that it favors certain methaphors, images, and ways of thinking.” [S. Papert]



- “A language that doesn't affect the way you think about programming is not worth knowing.” [A. Perlis]

High School CS

- Ministry of education committee
- Work on a new program 1990-1999.
  - Intensive work
  - Initially, studied the existing program and talked to many people
  - Decided on principles, basic format and main units
  - Formed additional teams to work on specific units – syllabus first, then teaching material.
- Basic view:
  - Don’t: Aim to train programmers.
  - Do: Produce informed adults



Real programmers of binary code

Principles of the program

- Prefer basic principles over technology that will become obsolete
- Algorithmic thinking
- Zipper principle: intertwine Concepts with their realization
- Balance concepts, challenges vs. pupils' abilities
- Modularity
- Required vs. Elective
- Require teacher training
- Require Labs

What I learned

- High School CS vs. College CS
  - Different goals
  - Different material
  - Different pedagogy
  - Different way of construction
  - But still influenced my thinking about CS college education
- Learned from the teachers
- No substitute to experimentation in the classroom

Other results of the Program

- Teachers
  - Past: many had no CS degree
  - Now: mostly CS graduates
- Having different groups develop each unit
  - Pluralism
  - Emergence of CS Education research
    - MSc, PhD students
    - Positions in Universities and Colleges

More thoughts

- Must be fun to learn
- “CS is lucky to be able to teach the newest”??
- Actually “CS is unlucky to be expected to teach the newest”

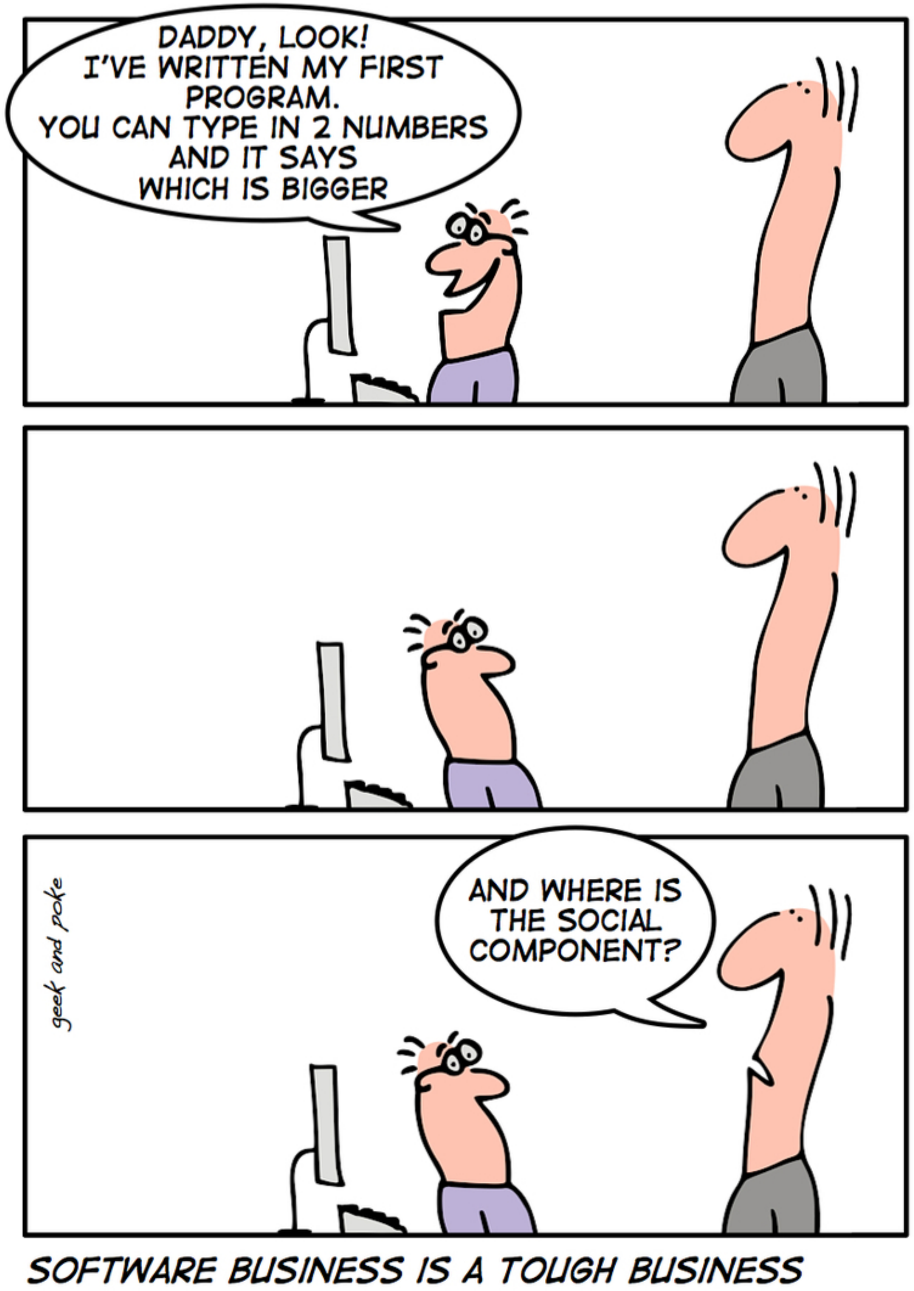
- New answer to “why study CS”:
  - Math is the queen and slave of the sciences
  - CS now has a similar role [C. Papadimitriou]
- The role of mistakes
  - The one who insists on never uttering an error must remain silent. [W. Heisenberg]

New CS High School program

- A few years ago, the new committee (headed by Judith Gal-Ezer, now by Shimon Schocken) decided to update the program
- Previously, language change
  - from Pascal, C to Java, C#
  - But no changes in content
- Now, update curriculum. Formed special Program Committee (headed by Orit Hazzan)
- Move to OO
- Other improvements in implementation
- Teachers working together on a pilot
  - proposing examples, discussion

CS in Junior High School? Earlier?

Should we teach programming?


DADDY, LOOK! I'VE WRITTEN MY FIRST PROGRAM. YOU CAN TYPE IN 2 NUMBERS AND IT SAYS WHICH IS BIGGER
AND WHERE IS THE SOCIAL COMPONENT?
geek and poke
SOFTWARE BUSINESS IS A TOUGH BUSINESS

CS in Junior High School? Earlier?

- Some initial study for Junior High School curriculum

- Beyond Computer literacy
- Three components:
  - Scratch
  - CS Unplugged
  - A simple computer simulator