High School Computer Science Teacher Preparation: Key Elements, Structure and Challenges

Orit Hazzan
Department of Education in Technology and Science
Technion - Israel Institute of Technology
Agenda

- High school computer science (CS) teacher preparation
  - From a model to its implantation

- Terminology:
  Didactics of Informatics = Computer Science Education
Collaborators

- Lenore Blum, Carnegie Mellon University, USA
- Judith Gal-Ezer, The Open University, Israel
- Tami Lapidot, Technion, Israel
- Noa Ragonis, Beit-Birl College, Israel
A Model for High School Computer Science Education

- **Israel**
  - 7 millions citizens
  - Central educational system
  - Worldwide center of hi-tech and innovation

- An advanced high school CS curriculum

  
  - The Israeli high-school CS curriculum is mentioned to illustrate that “the development of K–12 CS is making more headway internationally than in the US”.
A Model for High School Computer Science Education

Mandatory CS teaching license

CS teacher preparation programs

National CS curriculum and syllabus

Research in CS education

Connections between the model’s components

- Mandatory CS teaching license
- National CS curriculum and syllabus
- CS teacher preparation programs
- Research in CS education
How to initiate the model implementation?

- Mandatory CS teaching license
- CS teacher preparation programs
- National CS curriculum and syllabus
- Research in CS education
Overview

- Typical structure of a teacher preparation program in universities or colleges
  - A Bachelors degree in CS
  - Didactical courses (about one academic year)
    - General pedagogical courses (e.g., psychology)
    - Basic teaching skills
  - Specific pedagogical courses related to CS education
  - Methods of Teaching CS course
  - A practicum in real high school CS classes

- CS teacher preparation programs usually serve also in-service teachers
The Methods of Teaching CS Course
In many cases, methods courses are general pedagogical science teaching courses.

The Methods of Teaching CS Course

PCK: pedagogical-content knowledge (Shulman)

- How to make a subject understandable
- Students’ difficulties, preconceptions and misconceptions
- Strategies for coping with students’ misconceptions

Additional topics: Professional development, ethics, computer science

56-112 hours
The Methods of Teaching CS Course: Course objectives

1) Enhance prospective CS teachers' professional identity as CS teachers
   - What is computer science? Is computer science science?
   - Reflection and a reflective practitioner perspective (will be illustrated later)
   - Social issues: Ethics, diversity

2) Increase prospective CS teachers' awareness to the uniqueness of CS education
   - Build a test + checking guidelines
   - Mentoring software project development

3) Familiarize prospective CS teachers with the national CS curriculum
   - Exploration of the central ideas of the curriculum
   - Plan a study unit
   - Build a matriculation exam (details)
Build a matriculation exam

- **Cover** all subjects

- **Consider**:
  - different kinds of knowledge
  - different kinds of questions
  - the extent of each question
  - the test duration

- **Prepare** a checking guidance

- **Let** a colleague student to solve the test and **check** the test

- **Update** the test and the checking guidance accordingly
The Methods of Teaching CS Course: Course objectives

5) Expose prospective CS teachers to difficulties encountered by learners when learning different topics from the CS curriculum
   - Predict learners’ difficulties with respect to specific topics
   - Analyze learners’ (wrong and correct) answers

6) Enable prospective CS teachers to master pedagogical tools and teaching methods for teaching CS
   - Design a lab(-first) working sheet
   - Design a pedagogical game for teaching a specific CS topic
   - Develop a worksheet that uses the internet
The Methods of Teaching CS Course: Course objectives

7) Expose prospective CS teachers to the research in CS education and to its application in the teaching processes.
   - Read a research paper from the CS education research and explore its application to specific teaching situations
The Methods of Teaching CS Course: Course implementation principles

- Varied teaching methods; serve as a model
  - Active learning
    - Trigger
    - Hands-on tasks with the computer
  - Individual / pair / group work
  - Discussions
  - Reflections
The Methods of Teaching CS Course: Course implementation principles

- Multi-faceted exploration of central topics
  - teaching targets
  - concepts needed to be learnt
  - expected learners’ difficulties
  - lesson planning based on concepts needed to be learnt and the expected learners’ difficulties
    - the first lesson
    - sequence of lessons
  - kinds of questions for pupils’ assignments
  - kinds of questions for pupils’ evaluation
Illustration: Worksheet, work in pairs

A list of computer science concepts is given (in an alphabetical order)

- Abstraction
- Algorithm
- Assignment
- Boolean expression
- Control structures
- Correctness
- Data structure
- Debugging
- Efficiency
- Generalization
- Input-Output instructions
- Modularity
- Parameter
- Procedure
- Programming language
- Recursion
- Sorting
- Stepwise refinement
- Tracing
- Tree
- Variable

☐ Sort the above concepts into sets.
☐ Give a title to each set.
☐ To each set, add at least one concept.
A soft idea is a concept that can be neither rigidly nor formally defined, nor is it possible to guide students as to its precise application.

Computer science concepts are neither soft nor rigid.

Dijkstra's (1986) assertion: A computer scientist should move through many levels of abstraction, starting at the level represented by the machine and ending at the level of abstraction represented by the human thinking.
Teaching computer science soft ideas in the MTCS course

- Teaching CS soft ideas should not be neglected
- BUT: it is not a trivial matter to teach soft concepts

- Why to address the teaching of computer science soft ideas in the MTCS course?

- When to address soft ideas in the MTCS course?
Soft Ideas

- **Stage 2**: Construction of an activity that demonstrates one soft idea to computer science learners

- **Stage 3**: Construction of a question to be included in a test that checks learners' understanding of one soft idea
The prospective teachers' own understanding of CS soft ideas: What did they learn during their work on these activities? Did they improve their understanding of computer science soft ideas in general and of a specific computer science soft idea in particular? If yes — how? Do they still have questions about soft ideas? If yes — what kind of questions? What is the source of these questions?

The teaching of soft ideas in the high school: Should CS soft ideas be learned in high school? Possible difficulties that high school computer science students will face.
Soft ideas can be viewed as a meta-idea that is intertwined in teaching CS in general and in the MTCS course in particular with respect to different topics.
Practicum and Tutoring
Overview

- **Typical structure of a teacher preparation program in universities or colleges**
  - A Bachelors degree in CS
  - Didactical courses (about one academic year)
    - General pedagogical courses (e.g., psychology)
    - Basic teaching skills
  - Specific pedagogical courses related to CS education
    - Methods of Teaching CS course
    - A practicum in real high school CS classes

- **The focus of the talk**
  - CS teacher preparation programs usually serve also in-service teachers
Practicum in Real Classes

- Traditional practicum includes:
  - Observing some lessons of in-service CS teachers
  - Teaching several lessons by the prospective CS teachers

- The amount of hours dedicated for each activity varied between different institutes

- Guided by a university mentor and a high school teacher

- Reflective session after a lesson is taught by the prospective CS teacher
Tutoring

- **practicum** vs. **tutoring**

- Practicum: Integrated in the MTCS course

- Each prospective CS teacher = **tutor**
  tutors a CS novice student taking an
  introductory CS course = **tutee**

- The **tutors** are **new experts** in CS but
  **novice CS teachers**
Tutoring

- One-on-one tutoring sessions, 2 hours each, 5 each semester, 2 semesters
- The tutor guides the tutee through problem-solving processes
The tutors complete a tutoring session feedback worksheet after each session

- Contribution to the tutor's learning process
  - Tutors evaluate their own teaching and their tutee's understanding
  - The reflection fosters the need to plan the next session to meet the tutee's needs
Example of questions:

1) What concepts do you think constituted a difficulty for the tutee?
2) Describe the difficulty / misunderstanding / misconception..
3) What teaching tools did you use to help the student overcome the difficulty / misunderstanding / misconception..
4) What more would have helped you give the necessary assistance? (additional disciplinary knowledge, additional teaching knowledge, what kind of knowledge, which tools?)
5) If you could repeat this tutoring session, what would you do differently? (This question was suggested by one of the tutors)
Research Findings: Prospective CS Teachers as Reflective Practitioners

Eight viewpoints (VP) on reflection that the prospective CS teachers exhibited

- VP1. Reflection on learning CS in previous CS courses
- VP2. Reflection on learning in the MTCS course
- VP3. Reflection on teaching in the tutoring process
- VP4. Reflection on the tutee's learning processes
- VP5. Reflection on the encouragement of tutees to reflect during problem-solving processes
- VP6. Reflection as a tool to envision the tutors' future as CS teachers
- VP7. Reflection on the future pupils' understanding
- VP8. Meta reflection: Reflection on reflection processes
Illustration – Reflection on teaching in the tutoring process

- If I could repeat the tutoring session, I would give her additional time to think about the second part of the question and I would not give her such an obvious clue. I should have given her only a partial clue. [St. 7]

- I do not regret that we began the solution in a specific way and then switched to another way. I think that it exposed him [the tutee] to different thinking processes and to [the importance of] examining [different] ideas. [St. 19]
## Contribution of the reflection processes

(N=16, a 1-7 scale: 1-low, 7-high)

<table>
<thead>
<tr>
<th>Question</th>
<th>Average (SD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dealing with the different aspects of reflective thinking contributed to my learning.</td>
<td>5.88 (1.00)</td>
</tr>
<tr>
<td>I think that dealing with different aspects of reflective thinking will contribute to my work in the future.</td>
<td>6.26 (0.95)</td>
</tr>
<tr>
<td>It is important to complete a reflective report after each tutoring session.</td>
<td>6.39 (1.00)</td>
</tr>
</tbody>
</table>
The Model Application
How to initiate the model implementation?
How can the model be used by other countries?

**Mandatory CS teaching license**

**National CS curriculum and syllabus**

**Research in CS education**

**The problem:** How to establish a computer science teacher preparation program at your university?
How can the model be used by other countries?

- **The problem:** How to establish a computer science teacher preparation program at your university?

- **Possible solution:** The ECSTPP Workshop = Establishment of a Computer Science Teacher Preparation Program (ECSTPP) workshop

The ECSTPP Workshop - Population

- **Computer scientists** who
  - wish to establish a CS teacher preparation program in their universities
  - are not familiar with the practice of teaching CS in the high school and with CS education research

- **Designers of high school CS curricula** who
  - lack the background in CS education research
  - any university that wishes to establish a CS teacher preparation program is likely to recruit them to teach some of the courses in the program
The ECSTPP Workshop - Structure and contents

- Three consecutive stages
  - Common Ground stage
  - A three-day seminar
  - Action stage

- The Common Ground stage and the Action stage take place at the participants' institutions before and after the seminar, respectively
# The ECSTPP Workshop -
The schedule of the ECSTPP seminar

<table>
<thead>
<tr>
<th>#</th>
<th>Topic</th>
<th>Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Gathering, introduction and creating a community of learners</td>
<td>Introduction</td>
</tr>
<tr>
<td>2</td>
<td>The structure of a CS teacher preparation program</td>
<td>Introduction</td>
</tr>
<tr>
<td>3</td>
<td>The MTCS course – Part 1</td>
<td>MTCS course</td>
</tr>
<tr>
<td>4</td>
<td>Introduction to research in CSE</td>
<td>CSE Research</td>
</tr>
<tr>
<td>5</td>
<td>The MTCS course – Part 2</td>
<td>MTCS course</td>
</tr>
<tr>
<td>6</td>
<td>The MTCS course – Part 3</td>
<td>MTCS course</td>
</tr>
<tr>
<td>7</td>
<td>Research methods in CSE</td>
<td>CSE Research</td>
</tr>
<tr>
<td>8</td>
<td>The practicum – Part 1</td>
<td>Practicum</td>
</tr>
<tr>
<td>9</td>
<td>A reflective practitioner's perspective of CSE</td>
<td>CSE Research</td>
</tr>
<tr>
<td>10</td>
<td>The practicum – Part 2</td>
<td>Practicum</td>
</tr>
<tr>
<td>11</td>
<td>Action research</td>
<td>CSE Research</td>
</tr>
<tr>
<td>12</td>
<td>Launching the Action stage of the workshop</td>
<td>Integration of the 4 layers</td>
</tr>
</tbody>
</table>
Summary: High School Computer Science Teacher Preparation

- MTCS course
- Practicum, Tutoring
- ECSTPP workshop

CS teacher preparation programs

Mandatory CS teaching license

National CS curriculum and syllabus

Research in CS education
The Israeli model


The Method of Teaching CS course

Reference (cont)

- **Practicum**

- **Mentoring**
Reference (cont)

- **ECCTPP Workshop**

- **Book**

- **Additional resources**
  - Email: oritha@techunix.technion.ac.il
  - Web: http://edu.technion.ac.il/Faculty/OritH/HomePage/
Thank you!