

# **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

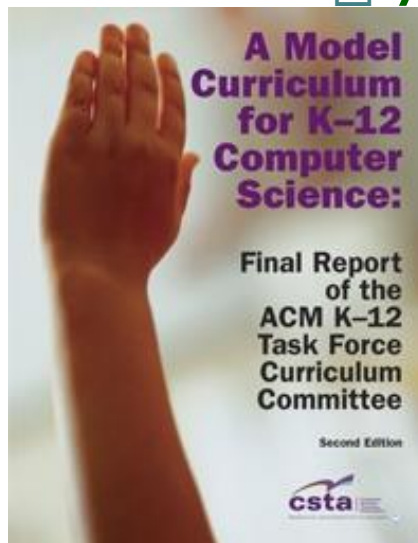
# A Model for High School Computer Science Education



## □ Israel

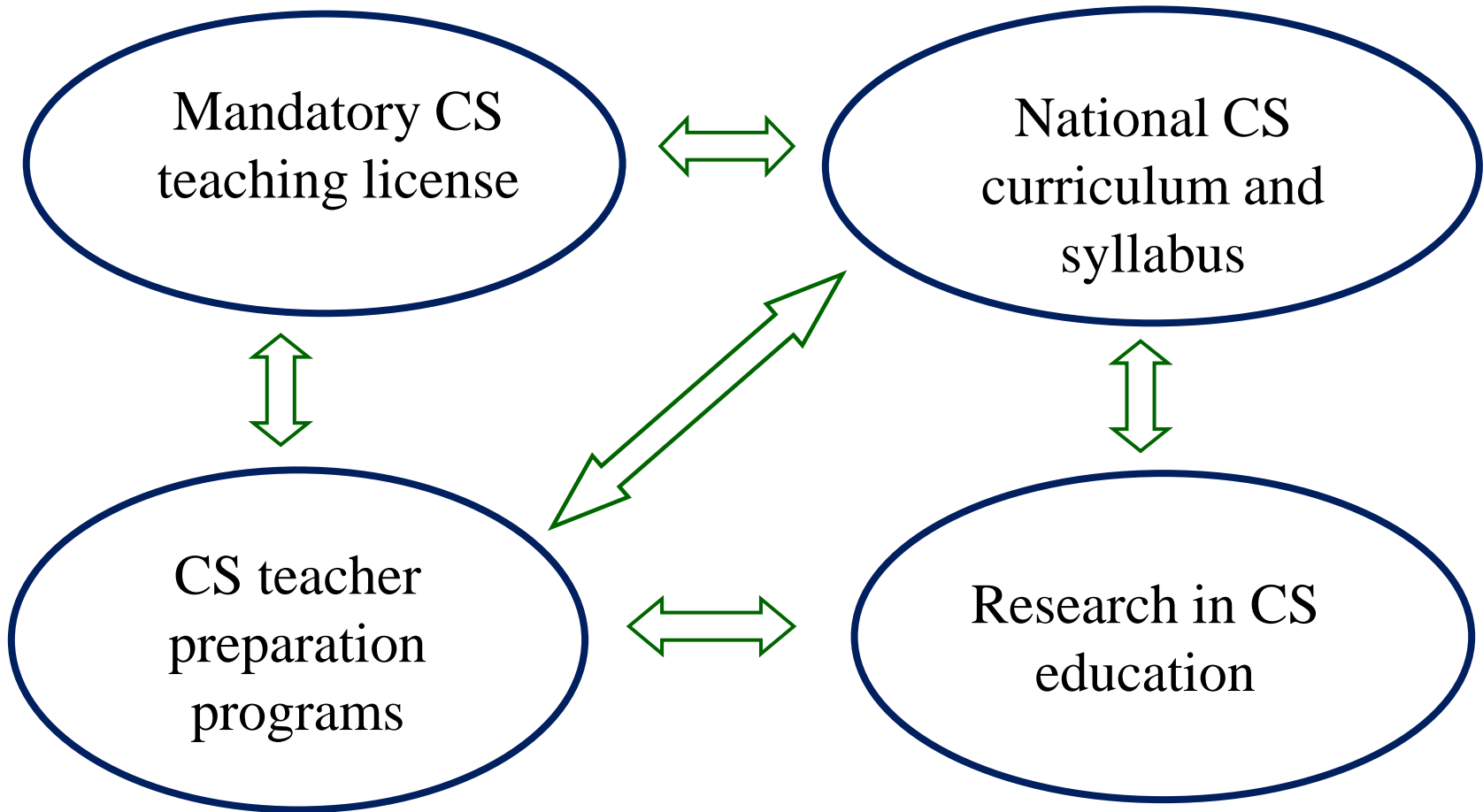
- 7 millions citizens
- Central educational system
- Worldwide center of hi-tech and innovation

## □ An advanced high school CS curriculum



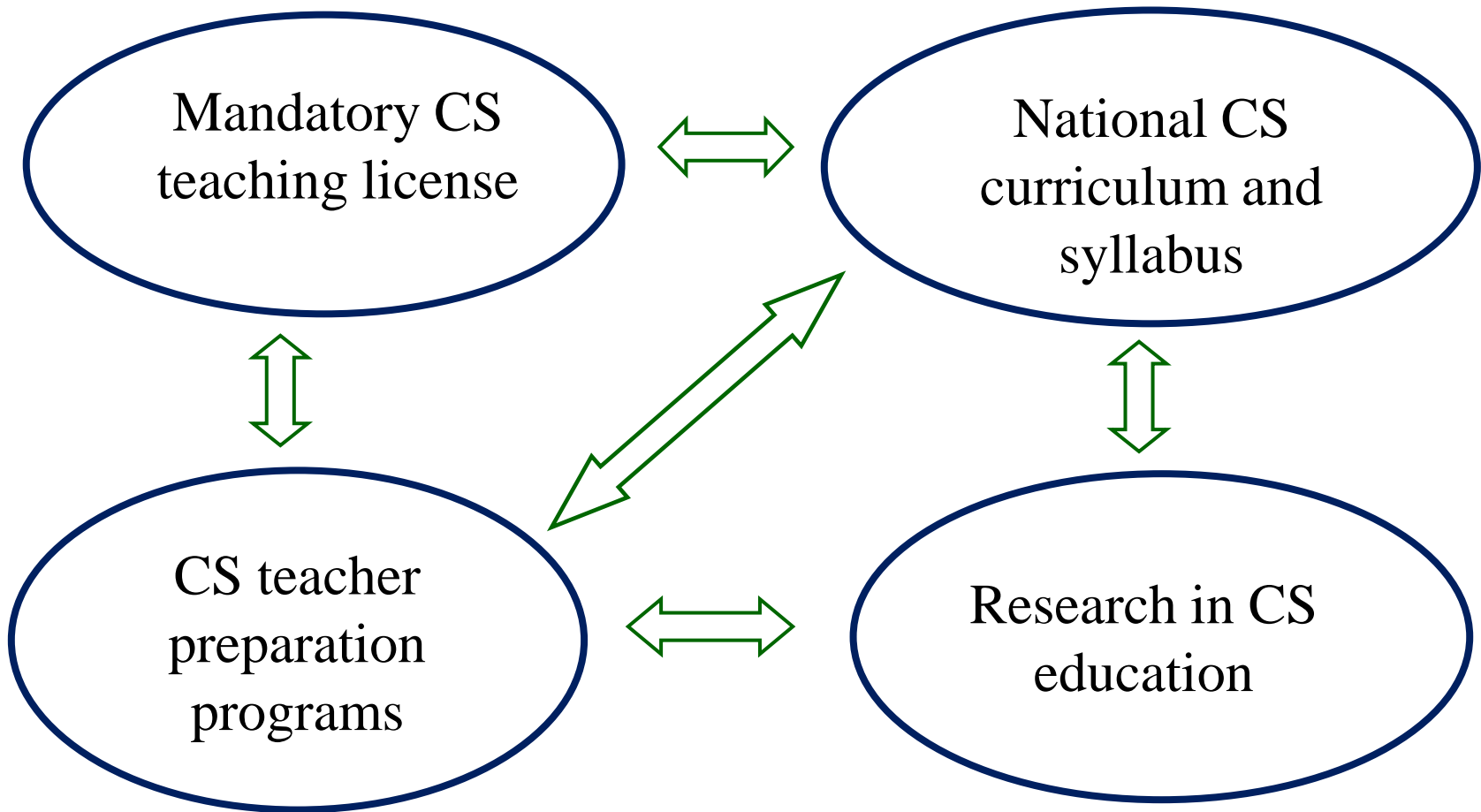
- Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., and Verno, A. (2003). ***A Model Curriculum for K-12 Computer Science. Report of the ACM K-12 Task Force Curriculum Committee.***
- The Israeli high-school CS curriculum is mentioned to illustrate that “the development of K–12 CS is making more headway internationally than in the US”.

# A Model for High School Computer Science Education

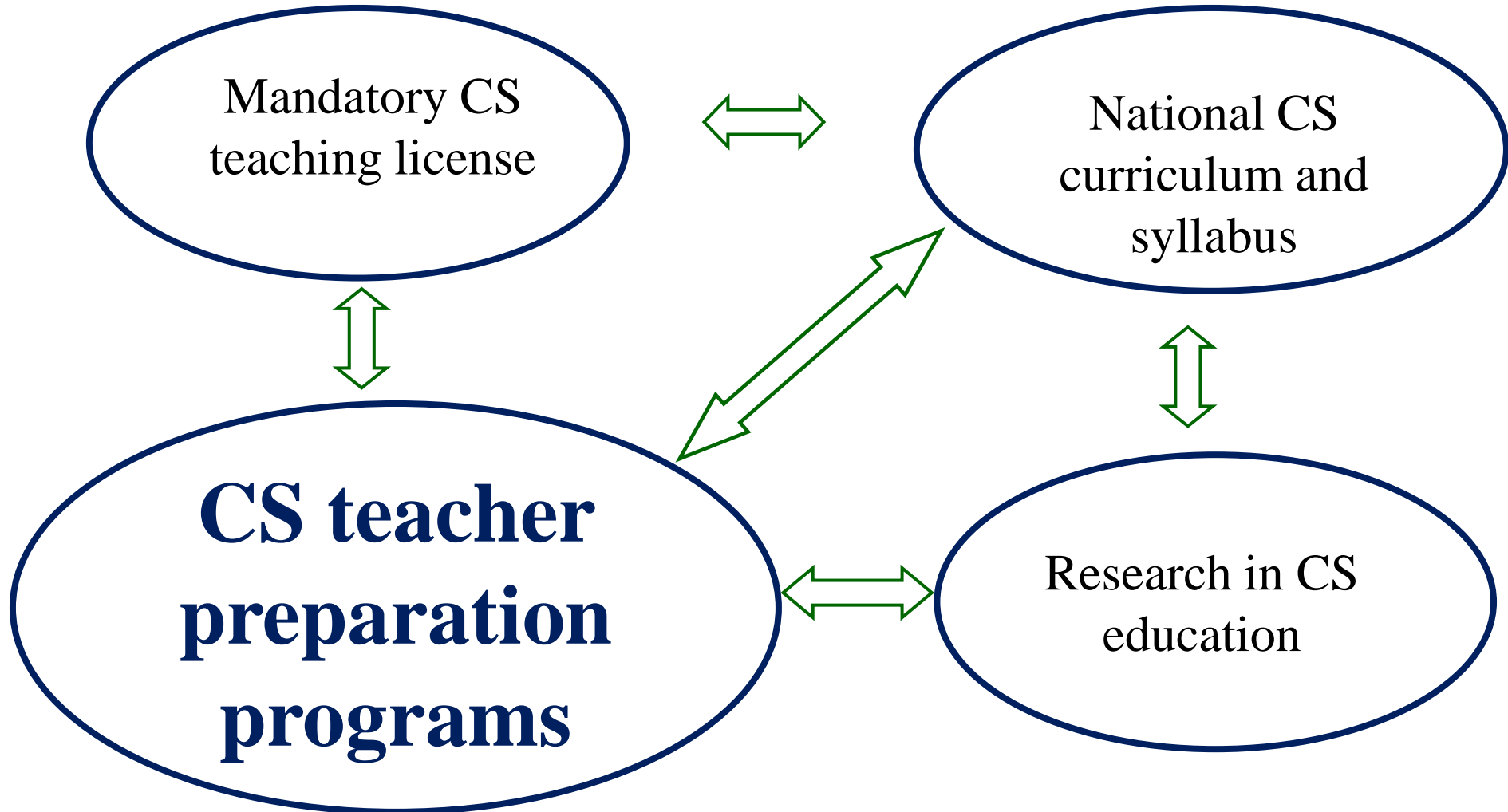


Hazzan, O., Gal-Ezer, J. and Blum, L. (2008). A model for high school Computer Science Education: The four key elements that make it!, *Proceedings of The 39th Technical Symposium on Computer Science Education*, Portland, Oregon, USA, pp. 281-285.

# Connections between the model's components




# How to initiate the model implementation?



# 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 focus  
of the talk

# The Methods of Teaching CS Course

# 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.

# Illustration: Computer Science Soft Ideas

- **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

# Soft Ideas- Stage 4 - Discussion

- **The prospective teachers' own understanding of CS soft ideas:** What did they learn during their work on the 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- Conclusion


---

*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**
- CS teacher preparation programs usually serve also in-service teachers



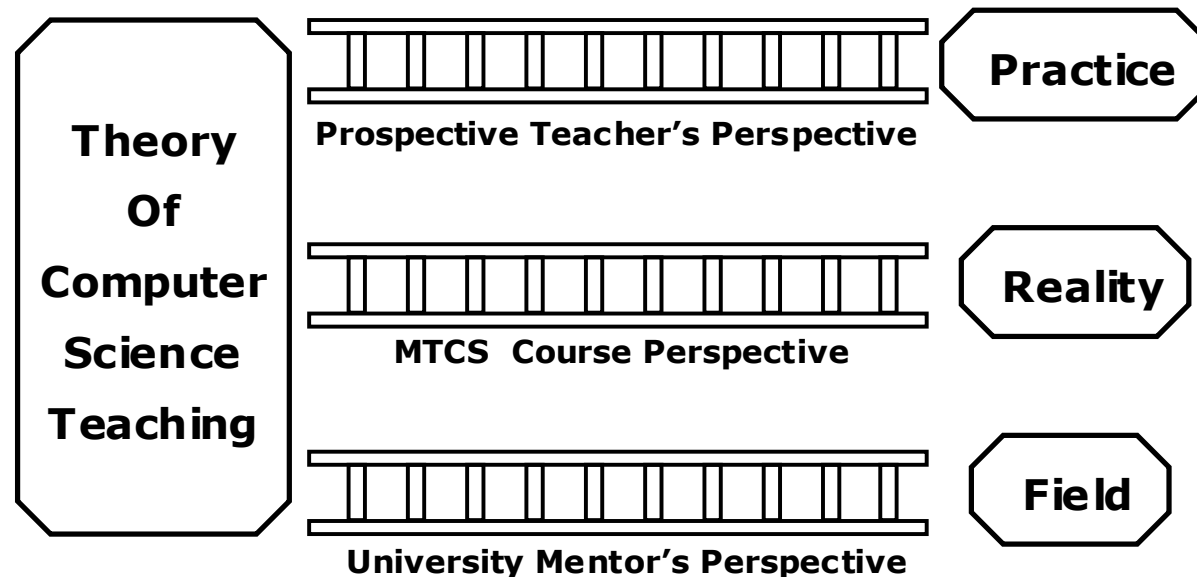
The focus  
of the talk

# 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

# Theory – Practice/Reality/Field Gap

## Bridging Gaps



Hazzan, O. and Lapidot, T. (2004). The practicum in computer science education: Bridging gaps between theoretical knowledge and actual performance, *inroads – the SIGCSE Bulletin* **36**(4), pp. 47-51.

# 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

# Reflection

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

# Tutoring session feedback worksheet

## □ 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]

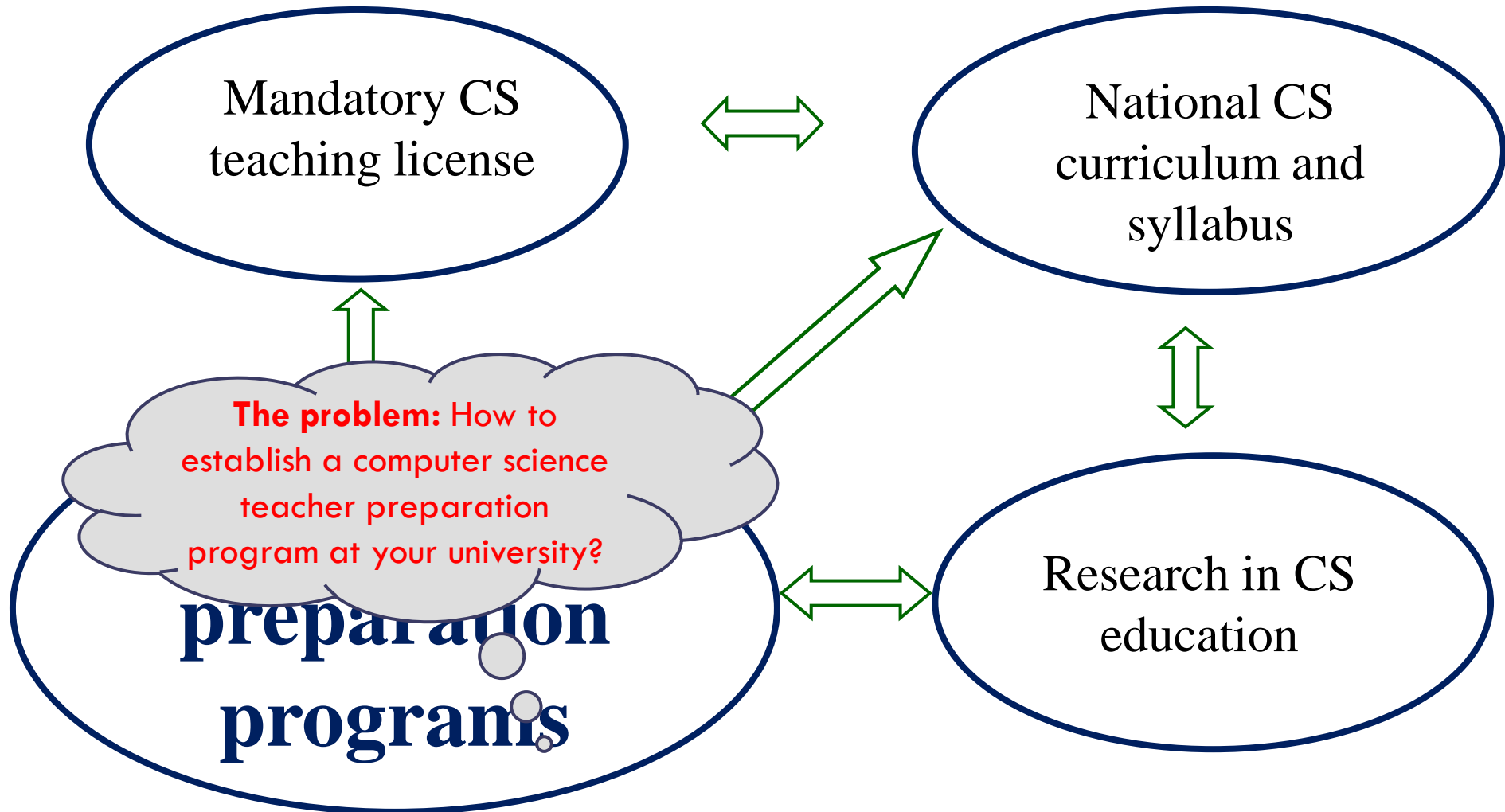
# Contribution of the reflection processes

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

Question	Average (SD)
<i>Dealing with the different aspects of reflective thinking contributed to my learning.</i>	5.88 (1.00)
<i>I think that dealing with different aspects of reflective thinking will contribute to my work in the future.</i>	6.26 (0.95)
<i>It is important to complete a reflective report after each tutoring session.</i>	6.39 (1.00)

# The Model Application

# How to initiate the model implementation? How can the model be used by other countries?



# 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
- Hazzan, O., Gal-Ezer, J. and Ragonis, N. (2010). How to establish a Computer Science teacher preparation program at your university? – The ECSTPP Workshop, *ACM Inroads* 1(1), pp. 35-39.

# 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

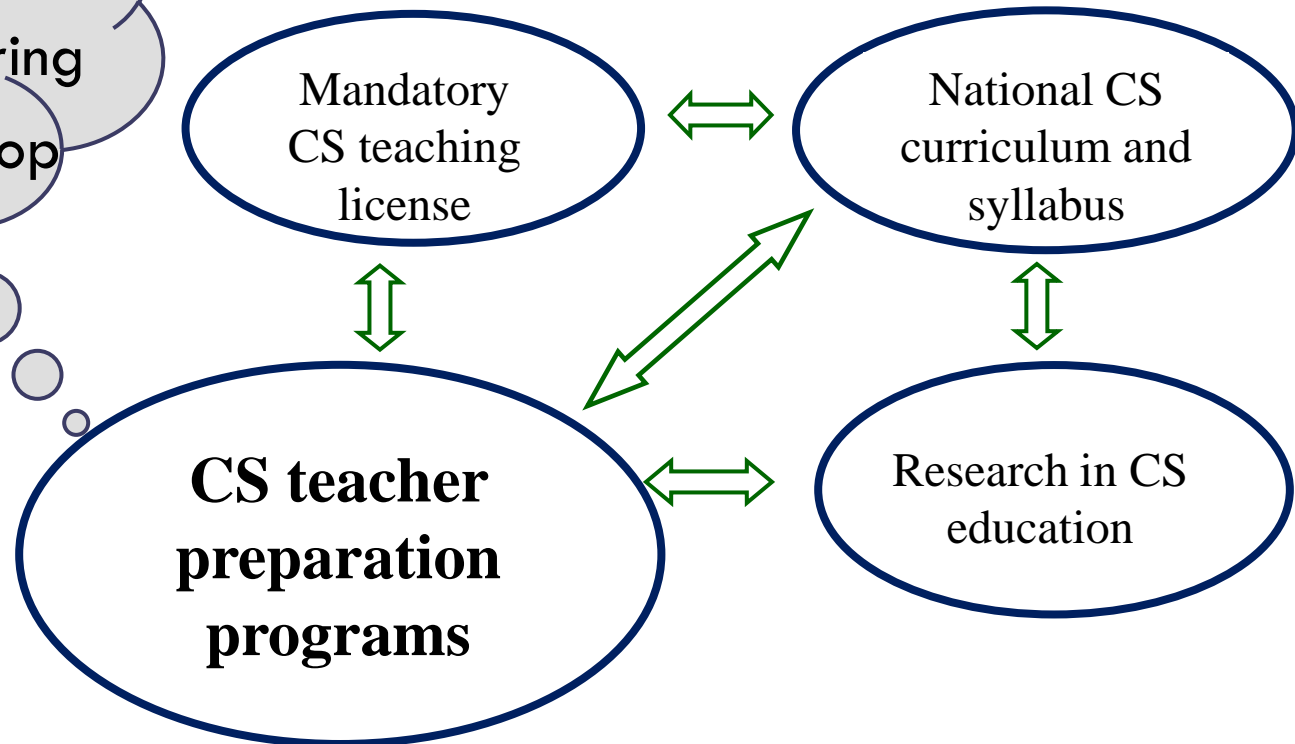
- **Three consecutive stages**
  - Common Ground stage
  - A three-day seminar
  - Action stage
- The Common Ground stage and the Action stage take place at the participants' institutions before and after the seminar, respectively

# The ECSTPP Workshop - The schedule of the ECSTPP seminar

#	Topic	Layer
1	Gathering, introduction and creating a community of learners	Introduction
2	The structure of a CS teacher preparation program	Introduction
3	The MTCS course – Part 1	MTCS course
4	Introduction to research in CSE	CSE Research
5	The MTCS course – Part 2	MTCS course
6	The MTCS course – Part 3	MTCS course
7	Research methods in CSE	CSE Research
8	The practicum – Part 1	Practicum
9	A reflective practitioner's perspective of CSE	CSE Research
10	The practicum – Part 2	Practicum
11	Action research	CSE Research
12	Launching the Action stage of the workshop	Integration of the 4 layers

# Summary: High School Computer Science Teacher Preparation

- ▣ MTCS course
- ▣ Practicum, Tutoring
- ▣ ECSTPP workshop



# Reference

## □ The Israeli model

- Hazzan, O., Gal-Ezer, J. and Blum, L. (2008). A model for high school Computer Science Education: The four key elements that make it!, *Proceedings of The 39th Technical Symposium on Computer Science Education*, Portland, Oregon, USA, pp. 281-285.
- Gal-Ezer, J., Hazzan, O. and Ragonis, N. (2009). Preparation of high school computer science teachers: The Israeli perspective, *The 40th Technical Symposium on Computer Science Education*, Chattanooga, TN, USE, pp. 269-270.

## □ The Method of Teaching CS course

- Lapidot, T. and Hazzan, O. (2003). Methods of Teaching Computer Science course for prospective teachers, *inroads – the SIGCSE Bulletin* **35**(4), pp. 29-34.
- Hazzan, O. and Lapidot, T. (2004). Construction of a professional perception in the "Methods of Teaching Computer Science" course, *inroads – the SIGCSE Bulletin* **36**(2). pp. 57-61.
- Lapidot, T. and Hazzan, O. (2005). Song debugging: Merging content and pedagogy in Computer Science education, *inroads – the SIGCSE Bulletin* **37**(4), pp. 79-83.
- Hazzan, O. and Lapidot, T. (2006). Social issues of Computer Science in the "Methods of Teaching Computer Science in the High School" course, *inroads – the SIGCSE Bulletin* **38**(2), pp. 72-75.
- Ragonis, N. and Hazzan, O. (2008). Disciplinary-pedagogical teacher preparation for pre-service Computer Science teachers: rational and implementation, *Informatics in Secondary Schools - Evolution and Perspective - ISSEP 2008, Lecture Notes in Computer Science*, Vol. 5090/2008, pp. 253-264.

# Reference (cont)

## □ Practicum

- Hazzan, O. and Lapidot, T. (2004). The practicum in computer science education: Bridging gaps between theoretical knowledge and actual performance, *inroads – the SIGCSE Bulletin* **36**(4), pp. 47-51.

## □ Mentoring

- Ragonis, N. and Hazzan, O. (2009). Integrating a tutoring model into the training of prospective Computer Science teachers, *The Journal of Computers in Mathematics and Science Teaching (JCMST)* **28**(3), pp. 309-339.
- Ragonis, N. and Hazzan, O. (2009). A tutoring model for promoting the pedagogical-disciplinary skills of prospective teachers, *Mentoring & Tutoring: Partnership in Learning* **17**(1), pp. 67 - 82.
- Ragonis, N. and Hazzan, O. (2008). Disciplinary-pedagogical teacher preparation for pre-service Computer Science teachers: rational and implementation, *Informatics in Secondary Schools - Evolution and Perspective - ISSEP 2008, Lecture Notes in Computer Science*, Vol. 5090/2008, pp. 253-264.
- Ragonis, N. and Hazzan, O. (2008). Tutoring model for promoting teaching skills of Computer Science prospective teachers, *The 13th Annual Conference on Innovation and Technology in Computer Science Education – ITiCSE 2008, Madrid, Spain*, pp. 276 - 280.
- Ragonis, N. and Hazzan, O. (2010). A reflective practitioner's perspective on computer science teacher preparation, *Proceedings of The 4th International Conference on Informatics in Secondary Schools: Evolution and Perspective (ISSEP)*, Zürich, Switzerland, pp. 90-106.

# Reference (cont)

## □ ECCTPP Workshop

- Hazzan, O., Gal-Ezer, J. and Ragonis, N. (2010). How to establish a Computer Science teacher preparation program at your university? – The ECSTPP Workshop, *ACM Inroads* 1(1), pp. 35-39.

## □ Book

- Hazzan, O., Lapidot, T. and Ragonis, N. (in preparation). *Guide to Teaching Computer Science*, Springer.

## □ Additional resources

- Email: [oritha@techunix.technion.ac.il](mailto:oritha@techunix.technion.ac.il)
- Web: <http://edu.technion.ac.il/Faculty/OritH/HomePage/>



**Thank you!**