Computational Thinking and Growth Mindset

Michael Lodi

PhD student, Dept. of Computer Science and Engineering
Alma Mater Studiorum - Università di Bologna, Italy
Visiting researcher (until April 2019)
University of Canterbury, Christchurch, New Zealand

`michael.lodi@unibo.it`

`michael.lodi@canterbury.ac.nz`

`lodi.ml`

30 November 2018

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Computational Thinking (CT)

- Computational thinking (Wing 2006) recognized "fundamental for everyone, not just computer scientists"
- Thinking like computer scientists to solve problems
- "The thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent." (Cuny, Snyder, and Wing 2010)

Origin and context (Lodi and Martini, under review)

- "Computational thinking" firstly used by (Papert 1980)
- In the context of constructionist learning theory
  - Constructivism: knowledge rediscovered/reconstructed rather than transmitted
  - Constructionism: learning through construction of meaningful (computational) artifacts
  - "Immerse yourself" in a world speaking that language (e.g. Mathland)
- Programming (LOGO) was one way to give intrinsic motivation, simulate every possibly significant world, etc.
- Papert focused on Math and Physics, but predicted "samba schools of computation": environments rich of computational principles and meaningful for the community
- Today, we should try to keep both of meanings (CS big ideas + meaningfulness and engagement for students)

The transfer problem (Lodi and Martini, under review)

- Papert misunderstood: never claimed programming "automatically" transfer in better learning
- Negative results about transfer obscured Papert's work
- Untested claims about CT transferring to all life contexts / transversal competences/skills
- Education research tells us transfer is difficult, especially between far domains (Ambrose et al. 2010)
- It can be achieved if taught explicitly (e.g. debug (Klahr and Carver 1988))

CT in K-12 education in Italy

- Countries introducing CS/CT in K-12 education - Italy too
- We push CT as a synonym for “CS core scientific concepts” to avoid misconceptions
- “Programma il Futuro” project endorsed by Ministry of Education (Programma il Futuro 2014-2018)
  - Italian translation of “Code.org” main courses
  - Support web site with video-tutorials for teachers, learning objectives...
  - Programming puzzles with increasing difficulty to learn programming concepts
- At the moment, only generic statement suggesting to teach “computational thinking” (described as a general problem solving tool, without reference to CS)
- Ongoing review of K-10 curriculum
- CINI (Informatics Interuniversity Consortium) proposed a curriculum with focus on principles rather than tools (encouraged for creative expression through computation) (Nardelli et al. 2017)

Teacher training

- Strong need for teacher training (In Italy, before y. 2000 you could become Primary Teacher right after High School; no CS neither in most of K-12 schools nor in Primary Teaching Degree)
- CS and programming associated with stereotypes (only for "geniuses", asocial male figures) (Lewis, Anderson, and Yasuhara 2016)
- Most (90%) of primary teachers are female, potentially more affected by stereotypes about CS

(Mis)Conceptions about CT and "coding"

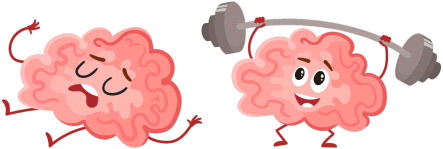
Sentiment and conceptions of 1000 PiF teachers.

- Perceived equal interest between boys and girls in Primary, decreasing in girls in Lower Secondary and even worse in Upper Secondary (Corradini, Lodi, and Nardelli 2017a)
- Ask for teacher training and more creative activities
- Only 1% gave a "good" definition of CT, only 10% an acceptable one (Corradini, Lodi, and Nardelli 2017b)
- All others had partial views (e.g. "general problem solving", "transversal competence") but lack fundamental elements (e.g. algorithms, programs or executor)
- 80% aware distinction between CT and "being able to use technology"
- Only 60% directly or indirectly related coding with computer programming (Corradini, Lodi, and Nardelli 2018)
- Conflicting misconceptions in the others: coding as "toy programming" vs. "more abstract and general"

Growth Mindset (GM) - “Tū whitia te hopo!”

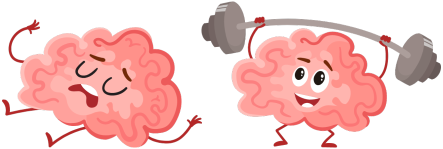
Solid cognitive theory by Carol Dweck (Dweck 2017)

- Different personal ideas about own intellectual abilities
  - Intelligence as a fixed trait (*fixed mindset*)
  - Intelligence can be developed (*growth mindset*)
- Growth mindset
  - behavior that fosters learning (asking questions, accepting challenges, deliberate practice, learning from critics, being inspired by others), in particular in STEM
  - protects women from negative stereotypes



Growth Mindset (GM) - “Tū whitia te hopo!”

- You can “teach” growth mindset
  - (Explicitly teaching it)
  - Giving “growth mindset” feedbacks
  - Teaching concrete strategies to cope with failures and errors
  - (About Math Growth Mindset) Teaching creative, open ended activities (e.g. projects) rather than mechanical exercises, teaching using visualization and connections between topics
- attention to “false” growth mindset
- teachers’ mindset is fundamental (primary teachers with math fear) (Dweck 2008)
- if you don’t change what/how you teach you’ll be pushing student to put effort in a transmissive/traditional education system



CS Growth mindset

- Different mindsets for different subjects
- (Little) research on relation between CS and GM found programming courses increase *fixed* mindset
- In facts, CS **requires** growth mindset (constantly faced with errors, need to work with peers) (Murphy and Thomas 2008)
- I think intrinsic characteristics of CS (e.g. open/real/authentic projects, iterative approach, debug, trial and error, collaboration rather than competition) can foster growth mindset...
- ... at least if we teach it as a creative subject!
- Unlikely to happen automatically (again, transfer is difficult)
- In other engineering fields, introducing open-ended design projects tended to lessen or eliminate the shift toward fixed mindset (Reid and Ferguson 2014)

Preliminary results - Growth mindset

- Measured GM level in female pre-service primary teachers before and after a “creative computing with Scratch” course - finding statistically significant, but little, increase in their growth mindset (Lodi 2018)
- High level of GM from the beginning (even if asked for specific CS GM, unpublished), probably due to:
  - Their field of study
  - Misunderstanding of the word “informatics”
  - Social desirability of self-reported levels of mindset

Ongoing data analysis - CS Growth Mindset

Analyzed (with questionnaires) GM and “CS GM” in high school (16 y.o.) students studying in CS or Chemistry or “Delivery&Logistics” focused tracks (in Italy you choose the kind of school and the track, not the single subjects).

- No significant difference in GM at the beginning and at the end of the school year
- No significant difference in “CS GM” in CS oriented classes
- Significant decrease of “CS GM” in non-CS classes (undesirable, if we think CS is valuable for all students)

Next steps

- Measuring GM through indicator in programming behaviors rather than with self-reported level of mindset
- Giving automated, context dependent, growth mindset feedback during a programming session (O'Rourke working on it)
- Design open-ended creative activities to teach "CS Big principles"

References I

Ambrose, Susan A et al. (2010). *How learning works: Seven research-based principles for smart teaching*. John Wiley & Sons.

Corradini, Isabella, Michael Lodi, and Enrico Nardelli (2017a). "Computational Thinking in Italian Schools: Quantitative Data and Teachers' Sentiment Analysis after Two Years of "Programma il Futuro" Project". In: *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '17. Bologna, Italy: ACM.

– (2017b). "Conceptions and Misconceptions About Computational Thinking Among Italian Primary School Teachers". In: *Proceedings of the 2017 ACM Conference on International Computing Education Research*. ICER '17. Tacoma, Washington, USA: ACM, pp. 136–144.

References II

- Corradini, Isabella, Michael Lodi, and Enrico Nardelli (2018). "An Investigation of Italian Primary School Teachers' View on Coding and Programming". In: *ISSEP*. Vol. 11169. Lecture Notes in Computer Science. Springer, pp. 228–243.
- Dweck, Carol S (2008). "Mindsets and Math/Science Achievement". In: *The Opportunity Equation*.
- Dweck, Carol S. (2017). *Mindset (Updated Edition) - Changing The Way You think To Fulfil Your Potential*. Robinson.
- Klahr, David and Sharon McCoy Carver (1988). "Cognitive objectives in a LOGO debugging curriculum: Instruction, learning, and transfer". In: *Cognitive Psychology* 20.3, pp. 362 –404. ISSN: 0010-0285. DOI: https://doi.org/10.1016/0010-0285(88)90004-7. URL: http://www.sciencedirect.com/science/article/pii/0010028588900047.

References III

Lewis, Colleen M., Ruth E. Anderson, and Ken Yasuhara (2016). ""I Don't Code All Day"". In: *Proceedings of the 2016 ACM Conference on International Computing Education Research - ICER '16*. DOI: 10.1145/2960310.2960332. URL: http://dx.doi.org/10.1145/2960310.2960332.

Lodi, Michael (2018). "Can Creative Computing Foster Growth Mindset?" In: *CC-TEL/TACKLE@EC-TEL*. Vol. 2190. CEUR Workshop Proceedings. CEUR-WS.org. URL: http://ceur-ws.org/Vol-2190/TACKLE_2018_paper_3.pdf.

Murphy, Laurie and Lynda Thomas (2008). "Dangers of a fixed mindset". In: *ACM SIGCSE Bulletin* 40.3, p. 271. ISSN: 0097-8418. DOI: 10.1145/1597849.1384344. URL: http://dx.doi.org/10.1145/1597849.1384344.

Nardelli, Enrico et al. (2017). *Proposal for a national Informatics curriculum in the Italian school*. Tech. rep. CINI. URL: https://www.consorzio-cini.it/images/PROPOSAL-Informatics-curriculum-Italian-school.pdf.

References IV

- Papert, Seymour (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. New York, NY, USA: Basic Books, Inc. ISBN: 0-465-04627-4.
- Programma il Futuro (2014-2018). *Main Site*. URL: http://programmailfuturo.it/.
- Reid, K. J. and D. M. Ferguson (2014). “Do design experiences in engineering build a growth mindset in students?” In: *2014 IEEE Integrated STEM Education Conference*, pp. 1–5. DOI: 10.1109/ISECon.2014.6891046.
- Wing, Jeannette M. (Mar. 2006). “Computational Thinking”. In: *Commun. ACM* 49.3, pp. 33–35.

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Think you! ;)

slides at `lodi.ml/talks`