**Background:** Since the reboot of Computer Science in the national curriculum in England, the teaching of Computer Science (CS) remains widely varied in quality. Many teachers continue to lack confidence to teach the subject and uptake by girls remains low (Royal Society, 2017). Nevertheless the CS curriculum contains a range of reasoning skills strongly desired by employers. Computational thinking has grown to consist of three main domains: computational concepts, computational practices, and computational perspectives (Brennan & Resnick, 2012). While these form a core focus of the CS curriculum, computational thinking skills, especially computational practices, can be useful to students in a range of classes within the curriculum (Selby, Dorling & Woollard, 2014). This paper explores to what extent GCSE and non-GCSE students differ in their computational concepts, practices and perspectives. Perhaps it is necessary to study CS to develop advanced competency in these key areas or perhaps they are so closely related to mathematical problem solving skills and approaches that there is no discernible difference between learners who study Mathematics alone and those who also additionally study Computer Science.

**Method:** 16 Year 11 students (aged 15-16) were drawn from two secondary schools in England. All participants were predicted to achieve high grades in their GCSE Mathematics course (grades 5-7). Of the CS sample 2 of 8 participants were female. Of the non-CS sample, 4 of 8 participants were female. These proportions were roughly in line with national participation rates (Royal Society, 2017). Participants were asked to asked to explain their approaches to a series of problems designed to explore their computational thinking skills in an extended interview, 'thinking aloud' to indicate their thought processes as they worked through the stages of each problem (Burke, 2012, Wang *et al*., 2012). Data was collected in field notes and audio recordings during the interviews and shared with the participants. These notes recorded both outcomes of the activity and on the thought processes which supported them. The categories identified in Brennan and Resnick's work (2012) were then applied in an iterative coding process.

**Results:** Results indicate that CS participants had considerably better computational concepts skills compared to non-CS participants. They were displayed better mastery of several CS practices: algorithmic thinking, formulation of a problem for a computer and evaluation of pre-existing code. However, there were a number of important areas in which CS-GCSE did not display discernible differences in their reasoning process compared to non-CS participants: abstraction, decomposition, generalizing/reusing, and logical reasoning. This suggests an overlap between CS thinking skills and mathematical thinking skills in some areas (Doyle, Stamouli & Huggard, 2005).

The CS-GCSE participants also demonstrated a greater willingness to try when engaging in the problems. CS participants regarded the computer as a tool to be programmed, rather than a sealed 'blackbox' - a common view among the non-CS group. These more open, proactive perspectives suggest that CS GCSE has the potential to positively affect the self-efficacy of learners in this area.

**Conclusion:** The findings of this study can add to the growing body of research informing the ongoing development of CS pedagogy. CPD providers can take up the Royal Society's challenge to develop relevant support which targets the gaps in teacher confidence and teacher competence in a focused manner. By acknowledging the overlap in some areas between thinking skills necessary for both general mathematical problem solving and for CS problem solving it will be possible to focus support in developing teacher competencies in those areas not already covered by mathematics: CS concepts, algorithmic thinking, problem formulation and computational perspectives.

**References:**

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada (pp. 1-25).

Burke, Q., (2012). The markings of a new pencil: Introducing programming-as-writing in the middle school classroom. Journal of Media Literacy Education, 4(2), 121-135.

E. Doyle, I. Stamouli and M. Huggard, (2005) "Computer anxiety, self-efficacy, computer experience: an investigation throughout a computer science degree," Proceedings Frontiers in Education 35th Annual Conference, Indianapolis, IN, 2005, pp. S2H-3. doi: 10.1109/FIE.2005.1612246

Royal Society, (2017). After the reboot: computing education in UK schools:

Selby, Cynthia, Dorling, Mark and Woollard, John (2014) Evidence of assessing computational thinking, University of Southampton

Wang, Y., Li, H., Feng, Y., Jiang, Y., & Liu, Y. (2012). Assessment of programming language learning based on peer code review model: Implementation and experience report. Computers & Education, 59(2), 412-422.