1-9-2019

Review of Measurements Used in Computing Education Research and Suggestions for Increasing Standardization

Lauren Margulieux
Georgia State University

Tuba Ketenci
Georgia Institute of Technology

Adriene Decker
Rochester Institute of Technology

Review of Measurements Used in Computing Education Research and Suggestions for Increasing Standardization

Lauren Margulieux[1], Tuba Ayer Ketenci[2], and Adrienne Decker[3]

[1]Department of Learning Sciences, Georgia State University

[2]School of Industrial & Systems Engineering, Georgia Institute of Technology

[3]Department of Engineering Education, University at Buffalo

Corresponding author: Lauren Margulieux, lmargulieux@gsu.edu, 404-413-8064

30 Pryor St SW, Atlanta, GA, 30302-3978

OrcID: 0000-0002-8800-2398

Abstract

The variables that researchers measure and how they measure them are central in any area of research. Which research questions can be asked and how they are answered depends on measurement. This paper describes a systematic review of the literature in computing education research to summarize the commonly used variables and measurements in 197 papers and to compare them to best practices in measurement for human-subjects research. Characteristics of the literature that are examined in the review include variables measured (including learner characteristics), measurements used, and type of data analysis. The review illuminates common practices related to each of these characteristics and their interactions with other characteristics. The paper lists standardized measurements that were used in the literature and highlights commonly used variables for which no standardized measures exist. To conclude, this review compares common practice in computing education to best practices in human-subjects research to make recommendations for increasing rigor.

Keywords: literature review, measurement, learner characteristics, standardized instruments, methods

Review of Measurements Used in Computing Education Research and Suggestions for Increasing Standardization

The field of computing education continually strives to increase the rigor and validity of its research (Almstrum, Hazzan, Guzdial, & Petre, 2005; Robins, 2015). To that end, systematic reviews of the literature help us to holistically consider the current state of the field and identify areas that could be improved (e.g., Lewis, Khayrallah, & Tsai, 2013; Lishinski, Good, Sands, & Yadav, 2016; Ramalingam & Wiedenbeck, 1998; Vihavainen, Airaksinen, & Watson, 2014). The current systematic literature review aims to serve the computing education community by examining the variables of learners and learning environments that we measure, the tools that we use to measure them, and the analyses that we use to process the data. The authors hope that this review provides computing education researchers with an array of commonly measured variables, commonly used measurements, and commonly practiced analyses to consider for future research.

Regardless of whether researchers' projects are qualitative, quantitative, or mixed methods and regardless of whether their projects are exploratory--confirmatory or design-focused--theory-focused, being aware of common measurement practices will allow researchers to take community practice into account when making decisions about measurement. Whether researchers choose to follow common practices or expand in novel directions—or ideally both—awareness of shared practices affords the community to build a more cohesive base of knowledge in computing education. To benchmark our practices to those outside of our community, this review concludes by comparing our practices for measurement to best practices in human-subjects research and highlighting potential areas for improvement. By consistently

adopting best practices, the computing education community's research can have higher impact within computing education and beyond to other education research communities.

Measurement is the method through which we collect data about the topic and variables that we are studying. In hard sciences, like physics, researchers develop tools to directly measure the factors in which they are interested, such as a scale to measure weight. In social sciences, like education, there is often not a direct method of measurement for the variables of interest. Take learning for an example. Many people measure learning through tests, but tests only approximate a person's knowledge. The same learner might perform differently on two tests about the same subject, depending on which questions are asked and how. In addition, the same learner might perform differently on the same test given at different times, depending on whether they are well-rested, anxious, etc. Therefore, how education researchers chose and implement measurements can have a meaningful impact on the validity and reliability of their findings.

Measurement has been addressed by several recent literature reviews and education policy updates. In K-12 education, the standards of research and evidence-based interventions are transforming to be more scientifically rigorous and transparent (Every Student Succeeds Act, 2015; National Board for Education Sciences, 2015). Within computing education, recent literature reviews and position papers have called for increased rigor in methodology (e.g., Lishinski et al., 2016). More specifically, papers that seek to aggregate data across multiple studies have decried the lack of standardization in variables that are measured and how they are measured (Decker et al., 2016; Decker et al., 2017; Ihantola et al., 2015; McGill et al., 2018).

This systematic literature review specifically aims to examine the standardization of measurement in computing education research. Standardization is valuable in fields that rely on human-subjects research because it affords comparisons among studies and improves

measurement tools by assessing their reliability and validity. Because the review evaluates standardization, it primarily discusses quantitative measures. This focus should not be mistaken to imply that quantitative measures are more valuable than qualitative measures or that standardization should be applied universally. Each of the authors personally use and advocate for using both qualitative and quantitative measures because both have unique values. Qualitative data are valuable, in part, because they are not standardized and, therefore, can more deeply and accurately describe learners and learning environments than most quantitative data, which attempts to represent people and places based on the same criteria and rated on a numeric scale. Therefore, for a paper about standardization, focusing on quantitative measures is more appropriate. Despite an emphasis on quantitative measurement, the authors explore the role of qualitative measurement and the conjunction of qualitative and quantitative data in computing education research. With this goal in mind, the authors had four main goals in this review:

- Describe the variables that are measured, including dependent/outcome variables and learner characteristics.
- Describe the common methods and instruments used to measure variables.
- Create a list of standardized instruments and make recommendations for additional variables can benefit from development of a standardized instrument.
- Identify variables that cannot reasonably be measured with standardized instruments and make recommendations for designing measurements and reporting on these variables.

The paper starts with a high-level primer of measurement. The primer is intended to explain important concepts of human-subjects measurement related to the current review, emphasizing topics that contribute to common mistakes. For those seeking additional

information, it can be found in the Research Methods Knowledge Base (Trochim, 2006); for those already familiar with the topics described in the headings, this section can be skipped. Next, the paper describes the research questions for the review and the methods used to answer the questions. Then, the next section summarizes the common practices of measurement as reported in 197 papers from *Computer Science Education*, *Transactions on Computing Education*, and proceedings of the *International Computing Education Research* conference. This section also identifies standardized instruments that are used in computing education research. Last, the paper concludes by comparing our community's common practices to best practices in human-subjects research to identify common practices that we should continue and practices that we should adopt more consistently.

1.1 Primer: Measurement in human-subjects research and common missteps

1.1.1 Types of measurement and common misconceptions

The two main categories of measurement are quantitative and qualitative. Quantitative measurement attempts to represent information numerically to make comparisons across participants easier and understand phenomena at the group level. In contrast, qualitative measurement represents information more descriptively and with less rigidity among participants to achieve a better understanding at the individual level. For example, to measure participants' self-efficacy, one could either ask learners to describe their level of self-efficacy with an open-ended question (qualitative), or one could ask learners to indicate, on a numeric scale, the degree to which they agree with statements about self-efficacy (quantitative). Neither approach is right or wrong, but which is more appropriate depends on the research goals. If you are gathering data from a large group of learners and self-efficacy is one of several individual differences considered, then quantitative measurement will allow you to easily compare across learners and

efficiently collect data. If you are exploring the effect of a new intervention on self-efficacy, then qualitative measurement will allow you to acquire a more nuanced understanding of the effect.

There are a couple of prevalent misconceptions about types of measurement. First, measurement is not necessarily the same as data. A researcher can use a qualitative measurement and then score it to produce quantitative data. A common example of this is giving students a problem for which they must write a program (qualitative measurement of learning) and then scoring it to provide a grade (quantitative data). In this case, the researcher could go back to the raw responses and look at a different aspect or score it for a different variable. This crossover can only be done with qualitative measurement; quantitative measurement will always provide immutable quantitative data.

The second misconception is that measurement and study design are dependent on each other. Due to this misconception people commonly assume that experimental research designs use quantitative measurement and non-experimental designs use qualitative measurements. In reality, researchers use both types of measurements, regardless of their research design, because each provides unique benefits.

1.1.2 Levels of measurement in quantitative data

Within quantitative data (not measurement), there are levels of measurement. Levels of measurement describes the relationship among different values of quantitative data. There are four levels of measurement.

- Nominal – names are swapped for values, and no relationship among values is implied, e.g., Java = 0, Python = 1, etc. Numeric values do not imply a relationship, e.g., Python is numerically higher, but not better, than Java.

- Ordinal – values are rank-ordered, but difference among values is not necessarily the same, e.g., for grades: C = 1, B = 2, A = 3, A is higher than B, and B is higher than C, but if students had a 91, 88, and 72 represented as A, B, and C, the difference between A and B is not necessarily the same as the difference between B and C.
- Interval – the difference among values is the same, but has no absolute zero (i.e., zero represents an absence of the variable being measured), e.g., the difference between 91 and 88 is the same as the difference between 88 and 85, but a grade of 0 does not mean that the student has no knowledge.
- Ratio – the difference among values is the same and has an absolute zero, which allows for mathematical relationships, e.g., for time on task: the difference between 1 and 2 hours is the same as the different between 2 and 3 hours, and 0 hours means no time was spent on the task. Due to the absolute zero, it is also valid to say that spending 2 hours on the task is twice as long as spending 1 hour.

Levels of measurement matter because they determine the statistical analysis that can be used on quantitative data. For example, Pearson's *r*, the most statistically powerful test for analyzing correlation, can be used for only interval or ratio data, while ordinal data must use Kendall's tau (τ) and dichotomous (nominal) data must use Spearman's rho (ρ). Higher levels of measurement (i.e., interval and ratio) are generally preferable because they provide more nuance in findings and allow for parametric tests. For example, when measuring self-efficacy, having a self-efficacy score between 0 and 10 for each learner provides more nuance than classifying each learner as high or low self-efficacy. Though it sometimes makes sense to convert interval or ratio data, such as a self-efficacy score, into ordinal data, such as high, medium, or low self-efficacy. If you make this conversion, it is ill-advised to use the **split-mean method**. The split-mean

method classifies participants with scores above the mean as “high” and those with scores below the mean as “low.” This results in participants with scores close to the mean (i.e., most participants in a normal distribution) to be grouped with participants furthest from the mean, even though participants close to the mean have more in common with each other than with those at the extremes, leading to unreliable results.

1.1.3 Validity

Measurement validity refers to whether you are measuring what you think you are measuring. Validity starts with how one operationalizes the variables that are measured. **Operationalization** converts the construct of interest, like knowledge or interest, into something that can be measured, like performance on a test or responses in an interview. Researchers should generally follow conventions around operationalization, such as measuring knowledge through an exam or measuring self-efficacy through a validated instrument, but they should also consider the limitations of operational definitions and how to overcome them. For example, does the exam that you give your students accurately represent programming knowledge that all students should have at that point in formal education? Does the validated instrument for self-efficacy that you are using accurately represent self-efficacy in the population that you are studying, or should you also include an interview? These questions represent threats to validity.

A full list of threats to validity can be found in the Knowledge Base (Trochim, 2006), but this paper will focus on two common threats to validity in computing education: self-report and evaluation apprehension. Self-report bias refers to participants wanting to represent themselves in a positive light or give the researchers what they want, causing them to skew the information that they provide. Use of self-report, whether qualitative or quantitative in nature, should be carefully considered (e.g., Rose, Porter, & Rogers, 2017). The other threat is evaluation

apprehension, in which participants are anxious about being evaluated and, therefore, act or perform differently than they normally would. One way to attempt to counteract evaluation apprehension is to tell participants, when appropriate, that the researchers are evaluating the instructions, materials, technology, etc., not the learners.

1.1.4 Reliability

Because we cannot directly measure many of the constructs that we are studying, most measurements that we use include some kind of error. Reliability is an assessment of how this error is likely to impact your results. Measurement reliability refers to the consistency of data that you would receive with a measurement. For example, if you gave the same participants the same tests graded by the same people in two parallel universes, then a highly reliable test would give you equivalent data for each participant. Reliability can refer to both the reliability of an instrument and the reliability of scoring data. The reliability of scoring data is commonly called interrater reliability. It refers to the consistency with which multiple people score (qualitative) data. There are different analyses to determine interrater reliability including Cohen's Kappa and intraclass correlation coefficient of consistency, ICC(C), or agreement, ICC(A). The standard procedure for a large amount of qualitative data is for everyone to score 20% of the data, assess reliability, and if it is sufficient, then to continue scoring with only one rater. Recent work has questioned whether 20% is sufficient in studies with small-to-medium sample sizes (Eagan et al., 2017).

A once-common measurement practice in education that has fallen out of favor is calculating difference scores for participants. **Difference scores** are calculated, for example, by subtracting a participant's score on a pre-test from their score on a post-test to represent the change in knowledge or any other variable. These calculations have fallen out of favor, however,

because they lead to unreliable data. In essence, a difference score collapses a pre-test score, which has its own error, with a post-test score, which also has its own error, into a new data point. In doing so, the difference score disregards these original sources of error so that statistical analyses cannot account for them, leading to a data point that is less reliable than the component scores.

In summary, without appropriate, valid, and reliable measurements, a field cannot be confident in its results and findings. The authors' goal for this paper was to review computing education research and identify the variables that researchers are using in their research and evaluate how they are measuring variables. By aggregating this data and analyzing trends, we aimed to compare practices in computing education research to best practices from other human-subjects research fields.

2 METHOD

This literature review follows the framework developed by Khan, Kunz, Kleijnen, and Antes (2003), with additional guidance from Petticrew and Roberts (2006). The framework has five foundational steps: frame the question, identify relevant work, assess the quality of the studies, summarize the evidence, and interpret the findings. This section describes the first three steps in detail, the last two steps are discussed in section 3.

2.1 Research Question

The overarching research question for this work was, in computing education research papers, 1) what variables are being measured, 2) using which instruments, and 3) with what types of data analyses? When considering the research question, the following basic characteristics were defined:

- Variables—Variables of interest to the research question as defined by the authors

    - Learner characteristics—A subgroup of variables that pertain to information collected about participants in computing education research studies that are not the main variables of interest
- Measurement—Characteristics of the measurements and instruments used
- Data analysis—Quantitative, qualitative, or mixed

2.2 Identification of relevant literature

To examine current practices for measurement and assessment in computing education research, papers published during 2013-2017 in *Computer Science Education* (CSE), *Transactions on Computing Education* (TOCE), and the International Computing Education Research (ICER) conference were analyzed. All articles from these sources, including those in special editions, were considered for inclusion except editorials. These sources were selected to represent the field for this review because they are peer-reviewed and in the top tier of scientifically rigorous journals and conferences in computing education, making them appropriate to examine the variables and instruments used in computing education research. General education journals that include computing education research were not included because the goal of the review was to evaluate measurement within the computing education research field, without the influence of other professional organizations and communities. The years 2013-2017 are included because they represent the most up-to-date work, especially the conference proceedings. For a field that is adapting and changing as quickly as computing education research, the recency of the publications best matched the goals of the review.

2.3 Selection of Studies and Assessing Quality

During 2013-2017 in CSE, TOCE, and ICER, 169 articles and 118 proceedings papers were published. Most of these papers described empirical work of human-subjects research that

measured variables in some form, and, therefore, are included in this review. Papers that were excluded (90) include review papers, evaluation papers (i.e., of a tool or of the state of computing education in a country), case studies (which are valuable but antithetical in purpose to standardization and thus not appropriate for this review), validation studies for developing a measurement, or, in two rare cases, because they did not describe measurement in enough detail to accurately classify them. This review, therefore, includes 197 empirical papers. Papers that described the validation of measurements, though not part of the review, are included in Table 3, which lists standardized instruments that are used in the community.

From each paper, the data for the collection points described in Table 1 were extracted via a careful analysis of the articles by the authors.

Table 1. Coding of relevant work

Characteristic	Collection points

Variables	• Variables of interest being measured
	• Learner and learning environment characteristics (prior experience, gender, age, etc.)
	  ○ Number of participants in study
	  ○ Age range or grade band
	  ○ Location/setting of research

Measurement	• Number and type of measurements used (survey, interview, test) • Use of standardized instruments
Data Analysis	• Quantitative, qualitative, or mixed

3 RESULTS AND DISCUSSION

This section aggregates the characteristics listed in Table 1 to describe trends from the 197 papers included in this analysis and considers interactions among characteristics.

3.1 Variables in computing education research and their measurement

The analysis identified the number of measurements that each paper reports and whether those measurements were qualitative, quantitative, or mixed. We found that the number of

measurements used depended on the analysis method. Papers that reported only qualitative analyses had an average of 1.2 measurements. Of 41 qualitative papers, 4 used two measurements, 3 used three measurements, and the rest used one measurement. This average and range is likely due to the longer nature of reporting qualitative data, which does not leave space for sufficiently reporting on multiple measurements. In addition, one measurement, such as an interview or artifact analysis, can produced a rich set of data when qualitatively analyzed, meaning multiple measures are not as important as in quantitative analyses for ensuring construct validity.

For papers that included quantitative analyses, the average number of measurements almost doubled. Papers that used mixed analyses had an average of 2.3 measurements. Of 64 mixed papers, 16 used only one measurement, 40 used 2-3 measurements, and 8 used 4-5 measurements. The 16 that used only one measurement analyzed the data both qualitatively and quantitatively. For a common example, after collecting data through a semi-structured interview, the researchers would quantitative code the interview and select quotes that provide higher explanatory value. Papers that reported only quantitative analysis had an average of 2.2 measurements. Of 92 quantitative papers, 39 used one measurement, 41 used 2-3 measurements, and 12 used 4-6 measurements. Those that used one measurement typically were measuring student performance through a complex measure, such as an exam with multiple components or a students' grade in a course. Those that used more than two measurements typically were measuring more than one variable of interest.

3.1.1 Variables of interest

Each of the measurements was coded for the variable that it was intended to measure based on the authors' descriptions. From post-hoc, preliminary, content analysis of these codes,

the authors identified nine categories of variables that were commonly used: performance, understanding, progress, experience, perceptions, collaboration, time, expert, and other (see Table 2). Based on these categories, two raters re-coded all measurements to fit into one of the categories. At the end of initial coding, with 96% agreement, codes not in agreement were resolved through discussion. Variables in the "other" category were reconsidered to identify common themes, but no sufficient commonalities were found. The other category included metrics that were specific for a particular study, such in the development of a model, or use of a tool, such as Scratch.

Table 2. Number of Metrics Used in CSE, TOCE, and ICER Papers '13-'17 Categorized by Type of Metric and Analysis.

	Analysis Type (total # of papers)		
Metric Type	Qualitative (41)	Quantitative (92)	Mixed (64)
Product Data*			
Performance	2 (5%)	60 (65%)	38 (60%)
Understanding	3 (7%)	17 (18%)	9 (14%)
Process Data			
Time	0 (0%)	14 (15%)	6 (9%)
Progress	10 (24%)	18 (20%)	19 (30%)
Experience	18 (44%)	24 (26%)	34 (53%)
Collaboration	3 (7%)	3 (3%)	6 (9%)
Other Data			
Perception	8 (20%)	23 (25%)	24 (38%)
Expert	7 (17%)	6 (7%)	2 (3%)
Other	1 (2%)	13 (14%)	5 (8%)

Note: The percentages represent the proportion of papers within an analysis type. For example, for performance metrics, 5% of qualitative papers included a performance metric, 65% of quantitative papers, and 60% of mixed papers. These percentages highlight whether a metric type was disproportionately used in certain analysis types.

The categories can be further abstracted into product data (i.e., data about the result of the learning process) and process data (i.e., data about the learning process; see Table 2). The papers

used two types of product data. The most common type of all metrics was performance. Performance was defined as the learner producing something, such as code or an artifact, to demonstrate the ability to apply knowledge. For example, grades on an exam, scores on problem solving tasks, and number of errors were all considered performance metrics. In total, 100 performance metrics were used in 95 papers, nearly half of all empirical papers in CSE, TOCE, and ICER over 5 years. The majority were used in quantitative or mixed analyses. Only two performance metrics were used in qualitative analysis, meaning that the other 39 papers that reported only qualitative analysis did not include a performance metric. This distribution represents common practice in education research that performance is quantified when analyzed and reported. In contrast, qualitative analysis often focuses on process data, such as students' motivation throughout a course or their thought processes while working on assignments.

Some of the performance metrics also included more conceptual-oriented questions to check for understanding, e.g., an exam might include conceptual and performance questions. If a metric included only conceptual-oriented questions, it was categorized as understanding. Understanding was defined as demonstrating conceptual knowledge without creating anything. For example, multiple choice questions about declarative knowledge and interviews about concepts were considered understanding metrics. Twenty-nine understanding metrics were used in 26 papers. Similar to the performance metrics, the majority of understanding metrics were used in quantitative or mixed analysis. Six of the papers that used an understanding metric also used a performance metric. Therefore, most papers that measured understanding did not measure performance. This decision would be appropriate for studies in which researchers were focusing on conceptual understanding, such as how algorithms work or what a computer sciences does, and did not expect that to affect performance or did not want to measure performance for the

participants' sake (e.g., researchers do not want to intimidate participants or take up more of their time or cognitive resources).

In total, 115 of 197 papers measured product data, either related to performance, understanding, or both. This result suggests that less than 60% of the research published in CSE, TOCE, or ICER between 2013 and 2017 measured learning outcomes, which is typical for education publications. Because education is about more than the product of learning, papers in education commonly do not report the products of learning and focus on the process of learning, especially in papers that use qualitative analysis. Qualitative analysis is much better suited to exploring the process of learning than the product of learning, at least within conventions of education research. In this review, at least 40% of the paper reported process or other, primarily perception, data without product data, which is in addition to the papers that included both process and product data.

The papers use four types of process data (see Table 2). Time was defined as time on task and is a direct measurement (not based on an operationalization) that yields quantitative data. Therefore, it is unsurprising that all measurements of time used quantitative or mixed analyses. Measuring time was not common. Perhaps it should become more common to provide valuable insight into the learning process.

The next type of process data was progress metrics. Progress was defined as describing the cognitive and behavioral process of learning or applying knowledge. For example, think aloud protocols, artifact analyses, help seeking behaviors, types of errors, and log data about number of submissions were all considered progress metrics. Forty-seven metrics were used in 44 papers, almost always in tandem with a performance or experience metric. A disproportionately large amount of papers that included a progress metric used mixed analysis,

reminiscent of common and highly valued practices in the learning sciences (Nathan & Sawyer, 2014). These types of papers tend to pair quantitative learning outcome data with qualitative data about the learning process to provide greater explanatory power for the learning outcomes and to examine learning trajectories, pain points, and effective activities.

In contrast to the cognitive and behavior aspects of progress metrics, experience metrics measured experiential and affective process data. For example, enjoyment of the task, motivation, engagement, and self-efficacy specifically for the task at hand (as opposed to general self-efficacy, which is a learner characteristic) were all considered experience metrics. Cognitive load was also considered an experience rather than progress metric because it was measured through self-report, and, therefore, describes the mental effort of learning, which is experiential. Seventy-six experience metrics were used in 68 papers. A disproportionately large amount was used in qualitative and mixed analyses. Qualitative data about experience can be extremely rich and useful to researchers, and experience can be difficult to capture when using quantitative metrics or quantifying qualitative data. If experience variables must be quantified, multiple or multifaceted metrics are common practice in social sciences to increase validity.

The other qualitative-focused, process metric was collaboration. Collaboration was defined as the social interactions of learners, either with other learners, instructors, or others. Though collaboration can be an extremely important aspect of the learning environment (Margolis, Estella, Goode, Holme, & Nao, 2008), it was measured only 12 times. Collaboration can be an omnipresent aspect of many learning environments, and peer learning activities, such as Peer Instruction (Lee, Garcia, & Porter, 2013) or peer dialogue (Asterhan & Schwarz, 2009; Smith et al., 2009), can greatly improve both student learning and experience. It is worth considering including more measurements of collaboration and considering the role of

collaboration in learning more often to thoroughly understand the learning environment. A tool that can help is Israel et al.'s (2016) C-COI observation instrument.

The other two categories that did not neatly fall into product or process data were perception and expert metrics. Perception was defined as learners' perceptions of computing as a field, of their performance on tasks, of their identities related to computing, and of their ability to complete tasks. This category included most attitude measurements, but some attitude measurements aligned better with the experience category based on the authors' reported intentions for the measurement. Fifty-six perception metrics were used in 54 papers, more than half of which were mixed methods. Like experience, perception can be problematic to distill into quantitative data, and qualitative or multifaceted quantitative data usually are the most useful for researchers. The last category, expert, was defined as information that only instructors or researchers would find valuable, such as pedagogical content knowledge or the learning trajectories of computing concepts. All 15 of expert metrics were used independently in 15 papers, and most were qualitative, which matches the nature of the expert knowledge that the papers were collecting.

3.1.2 Standardized instruments

The 37 standardized instruments that were used in the analyzed papers are listed in Table 3. This list is not a complete list of all standardized measures useful in computing education, only those used in the reviewed papers. Standardized instruments are valuable research tools because they have been evaluated for validity and reliability with a large sample of participants. In addition, using the same standardized tools across studies allows for comparisons among studies, which can develop knowledge in the field quickly. Direct measurements have the same features, and the reviewed papers included various direct measurements—log data, time on task,

GPS, number of errors, number of submissions, number of non-comment lines of code, word/character count, downloads per user, and eye tracking data.

Table 3: Standardized Instruments Used in CSE, TOCE, and ICER Papers '13-'17.

Learner Characteristic and Non-Computing Standardized Instruments			
Standardized Instrument	**Variable Intended to be Measured**	**Citation for Validation**	**Paper using Instrument**
Revised Purdue Spatial Visualization Test*	Spatial reasoning	Yoon, 2011	Cooper et al., 2015
Tuckman's Procrastination Scale*	Procrastination tendencies	Tuckman, 1991	Martin et al., 2015
Student Perceptions of Classroom Knowledge Building	Self-regulation strategy	Shell & Husman, 2008	Flanigan et al., 2015
Perceptions of Instrumentality Scale*	Instrumentality (motivation)	Husman et al., 2004	Peteranetz et al., 2016
Intrinsic Motivation Inventory*	Motivation	Deci & Ryan, 2003	Benhke et al., 2016
Project for the Analysis of Learning and Achievement in Mathematics	Career aspirations	Pekrun, Vom Hofe, Blum, Frenzel, Goetz, & Wartha, 2007	Peteranetz et al., 2016
Values Important to Career Selection Scale	Attitudes related to career aspirations	Lips, 1992	Beyer, 2014
Educational and Career Interest Scale	Educational and career interest in STEM	Oh et al., 2013	Ko & Davis, 2017
Mindset measurement*	Fixed vs. growth mindset	Blackwell, Trzesniewski, & Dweck, 2007	Stout & Blaney, 2017; Nelson et al., 2017
Epworth Sleepiness Scale*	Daytime chronic sleepiness	Johns, 1991	Nelson et al., 2017
Shapebuilder Task	Working memory capacity	Atkins et al., 2014	Margulieux & Catrambone, 2017
Cognitive Load Component Survey*	Cognitive load	Leppink et al., 2013	Morrison, 2017
Relational-Interdependent Self-Construal Scale*	Interpersonal orientation	Cross, Bacon, & Morris, 2000	Beyer, 2014
Motivated Strategies for Learning Questionnaire (MSLQ)*	Self-efficacy	Pintrich, Smith, Garcia, & McKeachie, 1993	Kurkovsky, 2013; Hundhausen et al., 2013

Approaches and Study Skills Inventory for Students	Learning strategies	Öhrstedt, 2009	Svedin & Bälter, 2016
Revised Study Process Questionaire (R-SPQ-2F)*	Students' approach to learning	Biggs, Kember, & Leung, 2001	Bati et al., 2014
Positive and Negative Affect Schedule (PANAS)*	Mood (positive and negative measured independently)	Watson, Clark, & Tellegen, 1988	Schneider et al., 2015
Big 5 Personality Inventory*	General personality profile	John & Srivastava, 1999	Beyer, 2014
Belbin Team Role test*	Collaborative role in teams	Belbin, 1993	Marshall et al., 2016
Myers-Briggs Type Indicator*	Personality profile	Myers et al. 1985	Theodoropoulos et al., 2017
Computing Standardized Measurement Instruments			
Standardized Instrument	**Variable Intended to be Measured**	**Citation for Validation**	**Paper using Instrument**
AP Computer Science A Test*	Generalized computer science knowledge	n/a, produced by College Board	Morrison et al., 2015
Fundamental CS1 Assessment (FCS1)	Language-independent assessment for introductory computer science	Elliot Tew & Guzdial, 2010	Parker et al., 2016
Second CS1 Assessment (SCS1)	Language-independent assessment for introductory computer science	Parker, Guzdial, & Engleman, 2016	Parker et al., 2016**; Nelson et al., 2017
Knowledge Test for CS1 Concepts	Programming knowledge (based on FCS1)	Lee & Ko, 2015	Lee & Ko, 2015
Misconceptions test	Misconceptions for first-year CS college course	Vahrenhold & Paul, 2014	Vahrenhold & Paul, 2014**
Basic Recursion Concept Inventory	Concept inventory for recursion topics	Hamouda et al., 2017	Hamouda et al., 2017**
Digital Logic Concept Inventory	Concept inventory for digital logic	Herman, Zilles, & Loui, 2014	Herman, Zilles, & Loui, 2014**; Ben-David Kolikant & Genut, 2017
Computing Attitudes Survey	Discipline-specific attitudes toward computing	Dorn & Elliot Tew, 2015	Dorn & Elliot Tew, 2015**; Majerko et al., 2016

Computer Programming Attitude Scale (CPAS)	Programming attitude scale for university students	Cetin & Ozden, 2015	Cetin & Andrews-Larson, 2016
Collaborative-Computing Observation Instrument	Collaboration in computing	Israel et al., 2016	Israel et al., 2016**; Israel et al., 2017
Programming self-efficacy measure	Programming self-efficacy	Ramalingam & Wiedenbeck, 1998	Nelson et al., 2017
First-year experience survey	Self-efficacy for first-year CS college students	Bhardwaj, 2017	Bhardwaj, 2017**
Self-efficacy for algorithms measure	Self-efficacy in introductory algorithms course	Danielsiek, Toma, & Vahrenhold, 2017	Danielsiek, Toma, & Vahrenhold, 2017**
Self-beliefs measure	Self-beliefs in CS1	Scott & Ghinea, 2014	Scott & Ghinea, 2014**
Computer Science Cognitive Load Component Survey	Cognitive load while completing computing activities	Morrison, Dorn, & Guzdial, 2014	Morrison, Dorn, & Guzdial, 2014**; Morrison et al. 2015; Harms et al., 2016; Margulieux & Catrambone, 2017
Computational Thinking Pattern Analysis	Coverage of patterns in CT inventory	Koh et al., 2014	Repenning et al., 2015; Basawapatna et al., 2013
ECS Assessment	Computational thinking in high school	Snow et al., 2017	Snow et al., 2017**

Note: * indicates an instrument that has been rigorously validated and widely used across multiple populations; ** indicates the initial validation paper was in the CSE, TOCE, and ICER papers reviewed.

Based on commonly measured variables that do not have standardized instruments, the community would benefit from the development and validation of a few new instruments. First, many of the non-computing standardized instruments have computing-specific equivalents (e.g., self-efficacy), but some do not and should, such as self-regulation strategies, career aspirations or interest, and mindset. We are not recommending that these variable should only be measured with standardized instrument. Instead, we are saying that adding a standardized instrument that

researchers can use in conjunction with other measurements would allow research results to be compared more easily. In addition, a standardized instrument for perceptions of the computing field (e.g., what is computing and what is computing used for) would likely be used often given that 54 papers measured this variable in one form or another but that not standardized instrument has been created for it. The last commonly used variable that has no standardized instrument is experience in paired programming. Paired programming is a common instructional method in computing, but few paper collected information about collaboration. Perhaps creating a standardized instrument to measure collaboration in paired programming would afford the community to more consistently explore collaborative aspects of the learning environment. This instrument would likely be most valuable as a self-report instrument rather than as an observation protocol so that it would scale better. Observation protocols for collaboration, especially when an observer can collect data from two people at a time, are better suited for qualitative data analysis and, therefore, benefit more from flexibility than from standardization.

3.1.3 Rigorous alternatives to standardized instruments

Though standardized instruments offer many benefits, developing and using them is not always appropriate or practical. For example, exploratory research should have a substantial unstandardized component because the goal is to explore a learning environment as it is. Unstandardized data collection methods allow for flexibility, though researchers might also collect some data with standardized instruments. Beyond exploratory research, using unstandardized instruments might still be a more valid decision because they can be better suited to a particular learning environment. For instance, using a test—an unstandardized instrument in the research community—to measure performance is better suited to a particular group of students than a standardized instrument like a concept inventory, though of course a researcher

could use both if time and energy allowed. In these cases, the community would benefit from describing unstandardized measures using common descriptors so that research can compared among the community more readily and accurately (i.e., so that we know whether we are comparing apples to apples or apples to oranges). McGill et al. (2018) provide valuable tips for reporting in computing education at https://csedresearch.org/guides/. Based on the current review, the authors would like to add to and double-down on items on this list.

For **exams and assignments**, the following properties should be reported:

- Proportions of question types (by points awarded or number of questions), including multiple choice, terminology, code tracing, code explaining, code writing, testing, or Parsons problems (e.g., Porter, Zingaro, & Lister, 2014).
- Transfer distance between instruction and student product, including isomorphic, contextual, or procedural (e.g., Morrison, Margulieux, & Guzdial, 2015).
- Alignment with Chi's ICAP framework (2009), including active, constructive or interactive tasks (e.g., Simon, Esper, Porter, & Cutts, 2013).
- Types of student mistakes (e.g., Brown & Altadmri, 2014; Hristova, Misra, Rutter, & Mercuri, 2003).
- How learners' code was scored (e.g., Fisler, 2014).
- Information about the distribution of scores, including mean, median, standard deviation, range, skewness, and kurtosis. This information can also include historical data from prior courses if the same assignments were used.
- Time on task or expected/allotted time on task. Similarly, reporting of course grades should include a description of the components of the grade, including the proportion attributed to homework, exams, projects, group work, etc.

Interviews were a common method of collecting data in the reviewed papers, but they were reported inconsistently. As a type of qualitative data collection, interview protocols should be flexible, but the following properties would be useful to report. 1) A priori questions or prompts distinguished from emergent questions or prompts. 2) Allotted time in addition to distribution of times, including mean, median, standard deviation, and range. 3) Coding scheme for analyzing results, including distinguishing between a priori and post hoc codes.

For analysis rather than measurement, some papers used standardized analysis protocols. Jadud's Error Quotient and/or the Watwin Score were used to predict student course performance with moderate success (Ahadi, Lister, Haapala, & Vihavainen, 2015; Jadud & Dorn, 2015). Carter et al. (2015) developed the Normalized Programming State Model to account for 41% of the variance in programming students' assignment grades and 26% of the variance in their final course grade. Fronza, Ioini, and Corral (2017) used McCabe's (1976) procedure to calculate cyclomatic complexity. Looking to the future, the field could greatly benefit from partially standardized practices for analyzing **log data**. Sixteen of the papers used log data for various purposes, and having common metrics (e.g., number of submissions or discussion posts) would help us to compare across studies.

3.2 Learner characteristic variables

In this section we discuss learner characteristics and their reporting by the studies. Standardized measures of learner characteristics are included in Table 3.

Learner characteristics are an important part of any education research paper. By describing the learner characteristics, researchers and educators can make judgements about whether the sample who completed the study matches their student population and how cautiously they should generalize the results. Despite this importance, only 49% of papers

reported learner characteristic information. In general, quantitative studies with large sample sizes (e.g., a large course or multiple courses) and qualitative studies with small sample sizes tended to not include demographic data. In both cases, collecting information about learners is important because with large sample sizes, the individual differences among students who receive the same instruction can be explored, and with small sample sizes, the participants tend to be less diverse and the results tend to be less generalizable.

Based on best practices in human-subjects research, the authors recommend starting or ending every study with a demographic questionnaire. Possible questions include common demographic data: gender, age, race/ethnicity, primary language, family annual income, academic major, enrollment status (part-time or full-time), and year in school or number of years/semester in college. The survey can also include information about extraneous variables that are relevant to success in computing education: GPA, prior experience in computing (including programming languages used), prior experience in math or science, expected grade in the course, and self-reported comfort with computers, self-efficacy, interest in computing, anxiety about computing, or expected difficulty of tasks (Rountree, Rountree, Robins, & Hannah, 2004). If possible, these self-reported variables should be measured with the validated measures, such as those in Table 3, but if full instruments are too long or unavailable, a question or two with a Likert-type scale can provide some insight.

Something that is rarely reported but should be regularly reported is information about instructors. Information about how long instructors have taught computing, what type of professional development or credentials they have, and what type of instruction and interactions with the learners they offer is valuable when assessing the generalizability of results. This is in line with the conclusions of McGill, Decker, and Abbott (2018), who were only looking at pre-

college interventions and noted as well that information about instructors is especially lacking in reporting. This work also provides recommendations for what types of data to collect and suggestions for how to accurately report on these demographic characteristics.

3.2.1 Common learner characteristics and measuring them

The most commonly reported learner characteristics were prior experience in computing (35, 18% of papers), gender (69, 35%), age (41, 21%; not including general grade band descriptors for the study like middle schoolers or CS1 students), and race (28, 14%). Prior experience was measured in several ways. Most commonly participants were asked in a survey whether they had prior experience with the task at hand, previous courses, or specific programming languages. In some cases, participants were asked to describe their prior experience in an open-ended question. In general, asking open-ended questions like this in a survey tends to provide unreliable data because participants are prone to skipping open-ended survey questions or not providing a comprehensive list. A more reliable way to collect data on prior computing experience is to ask specific questions about a specific task, course, or language. For more general questions about prior experience, researchers can ask students to indicate whether they have participated in informal experiences (e.g., museums, self-guided, after-school programs, or clubs) or formal experiences (e.g., courses or camps), how long they participated in each type, and when they participated (e.g., before 5th grade, middle school, high school, or college).

Gender can either be self-reported or experimenter/instructor-reported but be sure to say which. When asking participants to report their gender, current practice is, “What is your gender?” with options for “male,” “female,” “transgender,” “non-binary or agender,” and “prefer

not to answer." If the latter choices are not included, participants might skip this question, giving incomplete information, or select an inaccurate response.

To measure age, if possible, researchers should ask participants to report their age (e.g., 19) rather than ask them to select an age group (e.g., 18-20). Grouping ages provides ordinal data rather than interval data, restricting the type and power of analysis that researchers can conduct. As discussed in section 1.1.2, Pearson's *r* is used for only interval or ratio data. If you collect age groups, you must use Kendall's tau (τ), which has less statistical power.

Race is are best measured by self-report, but if self-report is impractical, estimates from an experimenter or instructor provide some information. The best survey question for ethnicity (in the United States) is check-all-that-apply question that asks, "Which ethnicities do you identify as?" with options for Asian, Black or African American, Native Hawaiian or Pacific Islander, Hispanic or Latinx, Middle Eastern, Native American, White or Caucasian, and Other (with a field to fill in).

Multiple choice or check-all-that-apply question formats are popular for learner characteristics because they are easy to quantify. When learner characteristics are not central to the research questions, they do not warrant qualitative coding by a researcher. In addition, the purpose of learner characteristic questions is often to group participants into general groups, so more nuanced measurement is not required. Other popular formats in human-subjects research for collecting data on learner characteristics are text boxes for numbers (e.g., GPA), drop-down menus, matrices of multiple choice or check-all-that-apply, Likert-type scales, and dichotomous questions (e.g., yes/no or true/false).

3.3 Quantitative and qualitative data and their relationship to participants

Of the 197 empirical papers included in the analysis, almost half (92) reported solely quantitative results. Some of these projects used qualitative measurements, such as code written by participants, but they reported only quantitative data analysis, such as percent correct or number of errors. These projects also collected quantitative data through surveys (typically Likert-type scales) or direct measurements (e.g., time on task). The other half of the papers was split between solely qualitative results (41) and mixed analyses (64). The qualitative projects generally reported interview, artifact analysis, and observation data. The mixed projects included all types of data collection and analysis including qualitative analysis of interviews or artifacts, quantitative analysis of qualitative data (e.g., open-ended surveys or test questions), and quantitative analysis of surveys and direct measurements. The quantitative, qualitative, and mixed studies were spread evenly throughout the five years, suggesting that this distribution is stable and that there are no trends, for example, towards more mixed method research.

3.3.1 Number of participants

The number of participants in each study, unsurprisingly, depended on the type of data that was analyzed (see Table 4). Studies with qualitative analyses only ranged from 2 to 33 participants with a mean of 19, a median of 17, and no outliers. On the other end of the spectrum, studies that used quantitative analyses had more participants. Studies ranged from 2 to 4068 participants with a mean of 401 and a median of 100. The distribution was normal between 23 and 388 participants but had outliers on both ends. The study with two participants, an eye-tracking study, was much below the normal distribution. On the upper end, there were five large studies that had between 621 and 1569 participants and two very large studies that had 3076 and 4068 participants. One study that used a database was excluded from these calculations because it included 32,680 participants, which is an extreme outlier.

Including both qualitative and quantitative analyses, studies that used a mixed approach had a middling number of participants. Studies ranged from 7 to 992 participants with a mean of 156 and a median of 58. The distribution was normal between 7 and 357 participants with five large study outliers, which had 501, 831, 961, 972, and 992 participants. In many of the larger sample size papers, qualitative data was not collected from all participants.

Almost 15% of papers did not explicitly give the total number of participants in a study, which is highly unusual in human-subjects research. This was more common in the ICER papers, with almost a quarter of papers omitting the total number of participants. Of the 30 papers that did not provide a total number of participants, 16 were quantitative, 5 were qualitative, and 9 were mixed. These papers were also distributed among the five years that were included in this analysis, suggesting that it is just as likely today to find a paper that does not provide the number of participants as it was five years ago. The community must be sure to report total number of participants in its papers, especially to aid consolidation efforts, such as effect size calculations and meta-analyses.

Table 4. Number of Participants and Research Settings in CSE, TOCE, and ICER Papers ’13-’17 Categorized by Type of Analysis.

	Analysis Type (total # of papers)		
	Qualitative (41)	Quantitative (92)	Mixed (64)
Number of Participants			
Range	2 - 33	2 - 4068	7 - 992
Mean	19	401	156
Median	17	100	58
School Research Settings			
Primary/K-5	2	1	1
Middle/6-8	2	3	2
High/Secondary/ 9-12	2	3	2
Summer camp	3	1	1
AP CS	1	1	1
College (CS1)	22 (6)	70 (36)	40 (8)

Adult/Teacher	6	4	7
Other/ Unspecified	3	9	10

3.3.2 Age of participants and research setting

The analyzed papers primarily focused on college-level computing education (see Table 4). The majority of research papers (69%) had college students as participants. The type of methods used in these papers were representative of the overall distribution, which is unsurprising given this group is the majority of papers. About a quarter of papers used qualitative methods (22), about half used quantitative methods (70), and about a quarter were mixed (40). Only 17 papers had adult participants, and 11 of those papers were with teachers. Papers with adult participants equally used qualitative (6), quantitative (4), or mixed (7) analyses.

The most common research setting was CS1 courses (i.e., college-level introductory programming courses). Of 197 papers, 50 of them were conducted in a CS1 course. Because these classes tend to have a lot of students and because most papers collected data in multiple CS1 courses, quantitative analysis were most common in this setting (36 used quantitative, 6 used qualitative, and 8 used mixed). In contrast to CS1 courses, other papers about college programming did not unrepresentatively favor quantitative analysis, including 4 CS0 courses (3 quantitative and 1 qualitative) and 13 advanced college programming courses (i.e., CS2 or higher, 5 quantitative, 1 qualitative, and 7 mixed). The remaining college-level studies did not specify in which course the research was conducted, or the research was not conducted in a course.

The remaining papers had participants from K-12. Four papers focused on primary school students, and they used qualitative (2), quantitative (1), and mixed (1) analyses. Seven papers focused on middle school students, using each qualitative (2), quantitative (3), and mixed (2)

analyses. The last seven papers focused on high school students, using each qualitative (2), quantitative (3), and mixed (2) analyses.

Research at the K-12 level, in contrast to the college level, favored mixed method research. Of 36 papers about computing units or activities in K-12, 8 used qualitative analysis, 13 used quantitative analysis, and 15 used both. Five papers about summer programs had 3 qualitative, 1 quantitative, and 1 mixed. Three papers about AP CS used one of each. The remaining 31 papers were conducted in research settings that either did not have enough cases to draw meaningful conclusions (e.g., boot camps, games, Scratch), were not restricted to a specific setting (e.g., models), or did not specify the research setting in detail.

4 CONCLUSIONS

Based on this review, computing education researchers use varied methods of collecting and analyzing data in human-subjects research with learners. Many people follow best practices that have been useful in related fields, such as the learning sciences. The best practices related to measurement that the community consistently uses include

- Collecting various types of data to match the research question, especially in studies (32%) that collected both qualitative and quantitative data.
- Collecting data through multiple measurements to create a more complete understanding of phenomena. Over half of papers (108) reported multiple measurements, and the papers that reported a single measure typically had a complex measurement that allowed for multiple analyses, such as a qualitative measurement or students' grades.
- Adopting standardized instruments from other fields when appropriate. Twenty of the 37 standardized instruments were developed outside of computing education.

- Developing and using standardized instruments to measure CS-specific common constructs. In addition to the 20 non-computing instruments, 17 computing-specific instruments have been developed or used in the papers that were reviewed.
- Having sufficient sample sizes for the type of data that is being analyzed. Median number of participants were generally sufficient for qualitative (17), quantitative (100), and mixed (58).

However, there are some best practices that the community does not consistently implement and that might hinder our progress in the coming years. These practices include

- Reporting number of participants and learner characteristics, especially prior knowledge and basic demographics, to improve readers' ability to assess research and aid later efforts to consolidate research studies. Only 85% of papers reported number of participants, and only 49% of papers reported learner characteristics. Both of these percentages should be about 100%.
- Measuring time on task when it does not disrupt the learning environment to give a more complete description of learning activities and account for differences in efficiency. Only 20 papers reported time on task, and the remaining 177 papers rarely described the amount of time that tasks should take or the amount of time allotted for them, which is useful in the case that it is impractical to collect data on time on task.
- Including detailed descriptions of the learning environment to provide information about the context of learning, including the resources available and the sociocultural environment. To this end, many researchers might find it prudent to include measurements of collaboration among their participants, even if collaboration is not part of their main research question. On 12 papers collected data about collaboration, and only

a handful of the remaining papers described collaboration among learners or interactions between students and instructors. Only rarely did papers describe the learning context in much detail, except that papers focusing on qualitative analysis were more likely to provide this information.

- Measuring both process and product data. About a third of paper reported both process and product data, and while it is sometime appropriate to collect one or the other, it is generally better to collect both to make research applicable to large range of readers. Data about the process of learning (e.g., progress or experience) provides insight into how learning outcomes develop and can be extremely valuable when paired with product data (i.e., performance or understanding). Fields that intentionally collect both types of data are becoming most influential in education practice and policy (Sommerhoff et al., 2018).

Though the community has adopted and developed a fair number of standardized instruments (apart from the instruments that were recommended to be developed), the vast majority of papers analyzed did not include standardized instruments, and many of them measured constructs that have a standardized instrument. There can be many barriers to using standardized instruments. The instrument might include more questions and take more time than the researcher wanted to devote. In response to this, the developers of instruments might consider developing a long and short version of the instrument and reporting the reliability and validity of each. The original developers do not have to complete this work. Any researchers who wanted a shorter version, if they had a large sample, could validate a short version of the instrument and compare it to the long version. Another barrier is that the instrument might not exactly measure the construct as the researcher intended. In response to this, the researcher should consider including both a standardized and non-standardized measurement. Then the study results can still

be compared to other studies, the non-standardized measurement can be compared to the standardized measurement, and the data would include the nuance that the researcher needs.

Despite the barriers, using standardized instruments has benefits. Using shared, validated measures would improve the validity and reliability of the communities' findings and help us to make quicker progress. In addition, when researchers use standardized instruments, they do not need to justify the reliability and validity of the measurement themselves. When standardized instruments are not feasible, following the suggested reporting recommendations will help the community to compare among studies better and build knowledge more systematically.

After touting the benefits of standardization, the authors would like to iterate that while the paper focuses on increasing standardization, they do not believe that standardization is the only way to increase rigor in computing education research nor that standardization should be the default goal when considering measurements. Especially in a relatively new field like computing education, we still have a lot left to explore, and unstandardized, and especially qualitative, measurements will help use retain authenticity as we explore new areas. The decision to use qualitative--quantitative and unstandardized--standardized measurements and analyses should be a decision based on the research questions and goals and the benefits of teach type of measurement. No single review could serve to focus on all of these benefits and tradeoffs at once, so this review focused on the benefits standardization at a time when the field is quickly growing.

By analyzing current practices and making recommendations for our future, this review aims to increase standardization, when appropriate, in computing education research. The authors hope that by considering measurement across the field, the paper demonstrates the importance of standard practices in collecting and reporting data to both authors and reviewers.

By using shared practices, we are better able to compare and consolidate research to more effectively speak to each other and to those outside of the field. Though standardization should not come at the cost of stifling creativity and nuance, being more mindful and intentional of standardization in computing education research has the potential to improve the influence of computing education research in practice and policy.

REFERENCES

Ahadi, A., Lister, R., Haapala, H., & Vihavainen, A. (2015, July). Exploring machine learning methods to automatically identify students in need of assistance. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research* (pp. 121-130). ACM.

Almstrum, V. L., Hazzan, O., Guzdial, M. & Petre, M. (2005). Challenges to computer science education research. Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '05). pp. 191–192. New York, NY: ACM.

Asterhan, C. S., & Schwarz, B. B. (2009). Argumentation and explanation in conceptual change: Indications from protocol analyses of peer-to-peer dialog. *Cognitive Science, 33*(3), 374-400.

Atkins, S. M., Sprenger, A. M., Colflesh, G. J., Briner, T. L., Buchanan, J. B., Chavis, S. E., ... & Harbison, J. I. (2014). Measuring working memory is all fun and games. Experimental Psychology.

Basawapatna, A. R., Repenning, A., Koh, K. H., & Nickerson, H. (2013, August). The zones of proximal flow: guiding students through a space of computational thinking skills and challenges. In *Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research* (pp. 67-74). ACM.

Bati, T. B., Gelderblom, H., & Van Biljon, J. (2014). A blended learning approach for teaching computer programming: design for large classes in Sub-Saharan Africa. *Computer Science Education, 24*(1), 71-99.

Behnke, K. A., Kos, B. A., & Bennett, J. K. (2016, August). Computer Science Principles: Impacting Student Motivation & Learning Within and Beyond the Classroom.

In *Proceedings of the 2016 ACM Conference on International Computing Education Research* (pp. 171-180). ACM.

Belbin, R. M. (1993). *Team Roles at Work*. Butterworth-Heinemann, Oxford, UK.

Ben-David Kolikant, Y., & Genut, S. (2017). The effect of prior education on students' competency in digital logic: the case of ultraorthodox Jewish students. *Computer Science Education*, *27*(3-4), 149-174.

Beyer, S. (2014). Why are women underrepresented in Computer Science? Gender differences in stereotypes, self-efficacy, values, and interests and predictors of future CS course-taking and grades. *Computer Science Education*, *24*(2-3), 153-192.

Bhardwaj, J. (2017). In search of self-efficacy: development of a new instrument for first year Computer Science students. *Computer Science Education*, *27*(2), 79-99.

Biggs, J., Kember, D., & Leung, D. Y. (2001). The revised two-factor study process questionnaire: R-SPQ-2F. *British Journal of Educational Psychology, 71*, 133–149.

Blackwell, L. S., Trzesniewski, K. H., & Dweck, C. S. (2007). Implicit theories of intelligence predict achievement across an adolescent transition: A longitudinal study and an intervention. Child Development, 78(1), 246-263.

Brown, N. C., & Altadmri, A. (2014, July). Investigating novice programming mistakes: Educator beliefs vs. student data. In Proceedings of the Tenth Annual Conference on International Computing Education Research (pp. 43-50). ACM.

Carter, A. S., Hundhausen, C. D., & Adesope, O. (2015, July). The normalized programming state model: Predicting student performance in computing courses based on programming behavior. In Proceedings of the Eleventh Annual International Conference on International Computing Education Research (pp. 141-150). ACM.

Cetin, I. (2013). Visualization: a tool for enhancing students' concept images of basic object-oriented concepts. *Computer Science Education, 23*(1), 1-23.

Cetin, I., & Ozden, M. Y. (2015). Development of computer programming attitude scale for university students. *Computer Applications in Engineering Education, 23*(5), 667-672.

Cooper, S., Wang, K., Israni, M., & Sorby, S. (2015, July). Spatial skills training in introductory computing. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research* (pp. 13-20). ACM.

Cross, S. E., Bacon, P. L., & Morris, M. L. (2000). The relational-interdependent self-construal and relationships. *Journal of Personality and Social Psychology, 78*, 791–808.

Danielsiek, H., Toma, L., & Vahrenhold, J. (2017, August). An Instrument to Assess Self-Efficacy in Introductory Algorithms Courses. In Proceedings of the 2017 ACM Conference on International Computing Education Research (pp. 217-225). ACM.

Deci, E. L., & Ryan, R. M. (2003). Intrinsic motivation inventory. Self-Determination Theory, 267.

Decker, A., & McGill, M.M. (2017) "Pre-College Computing Outreach Research: Towards Improving the Practice", *Proceedings of the 48th SIGCSE Technical Symposium of Computer Science Education,* March 8-11, 2017, Seattle, WA, pp. 153-158.

Decker A., McGill, M.M., & Settle, A. (2016) "Towards a Common Framework for Evaluating Computing Outreach Activities", *Proceedings of the 47th SIGCSE Technical Symposium of Computer Science Education,* March 2-5, 2016, Memphis, TN, pp. 627-632.

Dorn, B., & Elliott Tew, A. (2015). Empirical validation and application of the computing attitudes survey. Computer Science Education, 25 (1), 1-36.

Eagan, B. R., Rogers, B., Serlin, R., Ruis, A. R., Arastoopour Irgens, G., & Shaffer, D. W. (2017). Can We Rely on IRR? Testing the assumptions of inter-rater reliability. In International Conference on Computer Supported Collaborative Learning.

Ellitot Tew, A., & Guzdial, M. (2010, March). Developing a validated assessment of fundamental CS1 concepts. In Proceedings of the 41st ACM Technical Symposium on Computer Science Education (pp. 97-101). ACM.

Every Student Succeeds Act (2015). S.1177—114thCongress. Retrieved from https://www.govinfo.gov/app/details/CRPT-114hrpt354/CRPT-114hrpt354/context

Fisler, K. (2014, July). The recurring rainfall problem. In Proceedings of the Tenth Annual Conference on International Computing Education Research (pp. 35-42). ACM.

Flanigan, A. E., Peteranetz, M. S., Shell, D. F., & Soh, L. K. (2015, July). Exploring changes in computer science students' implicit theories of intelligence across the semester. In *Proceedings of the eleventh annual International Conference on International Computing Education Research* (pp. 161-168). ACM.

Fronza, I., Ioini, N. E., & Corral, L. (2017). Teaching computational thinking using agile software engineering methods: a framework for middle schools. *ACM Transactions on Computing Education (TOCE)*, *17*(4), 19.

Hamouda, S., Edwards, S. H., Elmongui, H. G., Ernst, J. V., & Shaffer, C. A. (2017). A basic recursion concept inventory. *Computer Science Education*, *27*(2), 121-148.

Harms, K. J., Chen, J., & Kelleher, C. L. (2016, August). Distractors in Parsons Problems Decrease Learning Efficiency for Young Novice Programmers. In *Proceedings of the 2016 ACM Conference on International Computing Education Research* (pp. 241-250). ACM.

Herman, G. L., Zilles, C., & Loui, M. C. (2014). A psychometric evaluation of the digital logic concept inventory. *Computer Science Education*, *24*(4), 277-303.

Hristova, M., Misra, A., Rutter, M., & Mercuri, R. (2003). Identifying and correcting Java programming errors for introductory computer science students. In Proceedings of the

Hundhausen, C. D., Agrawal, A., & Agarwal, P. (2013). Talking about code: Integrating pedagogical code reviews into early computing courses. *ACM Transactions on Computing Education (TOCE)*, *13*(3), 14.

Husman, J., Derryberry, W. P., Crowson, H. M., & Lomax, R. (2004). Instrumentality, task value, and intrinsic motivation: Making sense of their independent interdependence. Contemporary Educational Psychology, 29(1), 63-76.

Israel, M., Wherfel, Q. M., Shehab, S., Melvin, O., & Lash, T. (2017, August). Describing Elementary Students' Interactions in K-5 Puzzle-based Computer Science Environments using the Collaborative Computing Observation Instrument (C-COI). In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 110-117). ACM.

Israel, M., Wherfel, Q. M., Shehab, S., Ramos, E. A., Metzger, A., & Reese, G. C. (2016). Assessing collaborative computing: development of the Collaborative-Computing Observation Instrument (C-COI). Computer Science Education, 26(2-3), 208-233.

Jadud, M. C., & Dorn, B. (2015, July). Aggregate compilation behavior: Findings and implications from 27,698 users. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research* (pp. 131-139). ACM.

John, O. P., & Srivastava, S. (1999). The big five trait taxonomy: History, measurement, and theoretical perspectives. In L. A. Pervin & O. P. John (Eds.), *Handbook of personality: Theory and research* (2nd ed., pp. 102–138). New York, NY: Guilford Press.

Johns, M. W. (1991). A new method for measuring daytime sleepiness: the Epworth sleepiness scale. Sleep, 14(6), 540-545.

Khan, K.S., Kunz, R., Kleijnen, J., Antes, G. (2003). Five steps to conducting a systematic review. *Journal of the Royal Society of Medicine (JRSM).* 96(3), 118–121.

Ko, A. J., & Davis, K. (2017, August). Computing mentorship in a software boomtown: Relationships to adolescent interest and beliefs. In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 236-244). ACM.

Koh, K. H., Nickerson, H., Basawapatna, A., & Repenning, A. (2014, June). Early validation of computational thinking pattern analysis. In Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education (pp. 213-218). ACM.

Kurkovsky, S. (2013). Mobile game development: improving student engagement and motivation in introductory computing courses. *Computer Science Education*, *23*(2), 138-157.

Lee, C. B., Garcia, S., & Porter, L. (2013). Can peer instruction be effective in upper-division computer science courses?. *ACM Transactions on Computing Education (TOCE)*, *13*(3), 12.

Lee, M. J., & Ko, A. J. (2015, July). Comparing the effectiveness of online learning approaches on CS1 learning outcomes. In Proceedings of the Eleventh Annual International Conference on International Computing Education Research (pp. 237-246). ACM.

Leppink, J., Paas, F., Van der Vleuten, C. P., Van Gog, T., & Van Merriënboer, J. J. (2013). Development of an instrument for measuring different types of cognitive load. Behavior Research Methods, 45(4), 1058-1072.

Lewis, C. M., Khayrallah, H., & Tsai, A. (2013, August). Mining data from the AP CS A exam: patterns, non-patterns, and replication failure. In Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research (pp. 115-122). ACM.

Lips, H. M. (1992). Gender- and science-related attitudes as predictors of college students' academic choices. *Journal of Vocational Behavior*, *40*, 62–81. doi:10.1016/0001-8791 (92)90047-4

Lishinski, A., Good, J., Sands, P., & Yadav, A. (2016, August). Methodological Rigor and Theoretical Foundations of CS Education Research. In Proceedings of the 2016 ACM Conference on International Computing Education Research (pp. 161-169). ACM.

Magerko, B., Freeman, J., Mcklin, T., Reilly, M., Livingston, E., Mccoid, S., & Crews-Brown, A. (2016). Earsketch: A steam-based approach for underrepresented populations in high school computer science education. *ACM Transactions on Computing Education (TOCE)*, *16*(4), 14.

Margolis, J., Estella, R., Goode, J., Holme, J., & Nao, K. (2008). Stuck in the shallow end: Education, race, and computing. Cambridge, MA: MIT Press.

Margulieux, L., & Catrambone, R. (2017, August). Using Learners' Self-Explanations of Subgoals to Guide Initial Problem Solving in App Inventor. In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 21-29). ACM.

Marshall, L., Pieterse, V., Thompson, L., & Venter, D. M. (2016). Exploration of participation in student software engineering teams. *ACM Transactions on Computing Education (TOCE)*, *16*(2), 5.

Martin, J., Edwards, S. H., & Shaffer, C. A. (2015, July). The effects of procrastination interventions on programming project success. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research* (pp. 3-11). ACM.

McCabe, T. J. (1976). A complexity measure. IEEE Transactions on Software Engineering, (4), 308-320.

McGill, M. M., Decker, A., & Abbott, Z. (2018, February). Improving Research and Experience Reports of Pre-College Computing Activities: A Gap Analysis. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (pp. 964-969). ACM.

Morrison, B. B. (2017, August). Dual Modality Code Explanations for Novices: Unexpected Results. In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 226-235). ACM.

Morrison, B. B., Dorn, B., & Guzdial, M. (2014, July). Measuring cognitive load in introductory CS: adaptation of an instrument. In Proceedings of the 10th Annual Conference on ICER (pp. 131-138). ACM.

Morrison, B. B., Margulieux, L. E., & Guzdial, M. (2015, July). Subgoals, context, and worked examples in learning computing problem solving. In Proceedings of the Eleventh Annual International Conference on International Computing Education Research (pp. 21-29). ACM.

Myers, I. B., McCaulley, M. H., & Most, R. (1985). *Manual: A Guide to the Development and Use of the Myers-Briggs Type Indicator*. Consulting Psychologists Press.

Nathan, M. J., & Sawyer, R. K. (2014). Foundations of the learning sciences. In The Cambridge handbook of the learning sciences, 2nd ed. (pp. 21-43).

National Board for Education Sciences (2015). National Board for Education Sciences: 2014 NBES annual report, July 2013 through June 2014. Washington, DC.

Nelson, G. L., Xie, B., & Ko, A. J. (2017, August). Comprehension first: evaluating a novel pedagogy and tutoring system for program tracing in CS1. In *Proceedings of the 2017 ACM Conference on International Computing Education Research* (pp. 2-11). ACM.

Oh, Y. J., Jia, Y., Lorentson, M., & LaBanca, F. (2013). Development of the educational and career interest scale in science, technology, and mathematics for high school students. Journal of Science Education and Technology, 22(5), 780-790.

Öhrstedt, M. (2009). Approaches to studying, stress, academic achievement and the ability to assess own performance [Studieapproach, stress, studieresultat och förmåga att bedöma egen prestation]. (Master's thesis). Stockholm: department of psychology, Stockholm university.

Parker, M. C., Guzdial, M., & Engleman, S. (2016, August). Replication, validation, and use of a language independent CS1 knowledge assessment. In Proceedings of the 2016 Conference on ICER (pp. 93-101). ACM.

Pekrun, R., vom Hofe, R., Blum, W., Frenzel, A. C., Götz, T., & Wartha, S. (2007). Development of mathematical competencies in adolescence: The PALMA longitudinal study.

Peteranetz, M. S., Flanigan, A. E., Shell, D. F., & Soh, L. K. (2016, August). Perceived instrumentality and career aspirations in CS1 courses: Change and relationships with achievement. In *Proceedings of the 2016 ACM Conference on International Computing Education Research* (pp. 13-21). ACM.

Petticrew, M., & Roberts, H. (2006). Systematic Reviews in the Social Sciences: A practical guide. Oxford: Blackwell Publishing.

Pintrich, P. R., Smith, D. A., Garcia, T., & McKeachie, W. J. (1993). Reliability and predictive validity of the Motivated Strategies for Learning Questionnaire (MSLQ). Educational and Psychological Measurement, 53(3), 801-813.

Porter, L., Zingaro, D., & Lister, R. (2014, July). Predicting student success using fine grain clicker data. In Proceedings of the Tenth Annual Conference on International Computing Education Research (pp. 51-58). ACM.

Ramalingam, V., & Wiedenbeck, S. (1998). Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy. Journal of Educational Computing Research, 19(4), 367-381.

Repenning, A., Webb, D. C., Koh, K. H., Nickerson, H., Miller, S. B., Brand, C., ... & Repenning, N. (2015). Scalable game design: A strategy to bring systemic computer science education to schools through game design and simulation creation. *ACM Transactions on Computing Education (TOCE)*, *15*(2), 11.

Rich, K. M., Strickland, C., Binkowski, T. A., Moran, C., & Franklin, D. (2018). K-8 learning trajectories derived from research literature: sequence, repetition, conditionals. ACM Inroads, 9(1), 46-55.

Robins, A. (2015). The ongoing challenges of computer science education research. Computer Science Education, 25(2):115-119.

Rosen, J. A., Porter, S. R., & Rogers, J. (2017). Understanding student self-reports of academic performance and course-taking behavior. *AERA Open*, *3*(2), 1-14.

Rountree, N., Rountree, J., Robins, A., & Hannah, R. (2004). Interacting factors that predict success and failure in a CSI course. *SIGCSE Bulletin, 33*(4), pp 101-104.

Schneider, K., Liskin, O., Paulsen, H., & Kauffeld, S. (2015). Media, mood, and meetings: related to project success?. *ACM Transactions on Computing Education (TOCE), 15*(4), 21.

Scott, M. J., & Ghinea, G. (2014, July). Measuring enrichment: the assembly and validation of an instrument to assess student self-beliefs in CS1. In Proceedings of the Tenth Annual Conference on International Computing Education Research (pp. 123-130). ACM.

Shell, D. F., & Husman, J. (2008). Control, motivation, affect, and strategic self-regulation in the college classroom: A multidimensional phenomenon. Journal of Educational Psychology, 100(2), 443.

Simon, B., Esper, S., Porter, L., & Cutts, Q. (2013, August). Student experience in a student-centered peer instruction classroom. In Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research (pp. 129-136). ACM.

Smith, M. K., Wood, W. B., Adams, W. K., Wieman, C., Knight, J. K., Guild, N., & Su, T. T. (2009). Why peer discussion improves student performance on in-class concept questions. *Science, 323*(5910), 122-124.

Sommerhoff, D., Szameitat, A., Vogel, F., Chernikova, O., Loderer, K., & Fischer, F. (2018). What do we teach when we teach the learning sciences? A document analysis of 75 graduate programs. Journal of the Learning Sciences.

Snow, E., Rutstein, D., Bienkowski, M., & Xu, Y. (2017, August). Principled Assessment of Student Learning in High School Computer Science. In Proceedings of the 2017 ACM Conference on International Computing Education Research (pp. 209-216). ACM.

Stout, J. G., & Blaney, J. M. (2017). "But it doesn't come naturally": how effort expenditure shapes the benefit of growth mindset on women's sense of intellectual belonging in computing. *Computer Science Education*, *27*(3-4), 215-228.

Svedin, M., & Bälter, O. (2016). Gender neutrality improved completion rate for all. *Computer Science Education*, *26*(2-3), 192-207.

Theodoropoulos, A., Antoniou, A., & Lepouras, G. (2017). How do different cognitive styles affect learning programming? Insights from a game-based approach in Greek schools. *ACM Transactions on Computing Education (TOCE)*, *17*(1), 3.

Trochim, W. M. (2006). The Research Methods Knowledge Base, 2nd Edition. Retrieved from http://www.socialresearchmethods.net/kb/.

Tuckman, B. W. (1991). The development and concurrent validity of the procrastination scale. Educational and Psychological Measurement, 51(2), 473-480.

Vahrenhold, J., & Paul, W. (2014). Developing and validating test items for first-year computer science courses. *Computer Science Education*, *24*(4), 304-333.

Vihavainen, A., Airaksinen, J., & Watson, C. (2014, July). A systematic review of approaches for teaching introductory programming and their influence on success. In Proceedings of

the Tenth Annual Conference on International Computing Education Research (pp. 19-26). ACM.

Watson, D., Clark, L. A., & Tellegen, A. (1988). Development and validation of brief measures of positive and negative affect: the PANAS scales. Journal of Personality and Social Psychology, 54(6), 1063.

Yoon, S. Y. (2011). Psychometric properties of the revised Purdue spatial visualization tests: visualization of rotations (The Revised PSVT: R). Purdue University.