

Validation of a Parallel and Distributed Computing Instrument for Early Computing Research

1st Chris Bourke
School of Computing
University of Nebraska–Lincoln
Lincoln NE, USA
chris.bourke@unl.edu
0000-0002-1626-054X

2nd David P. Bunde
Computer Science
Knox College
Galesburg IL, USA
dbunde@knox.edu
0000-0001-6334-356X

3rd Charlotte Gruner
Computer Science
Casper College
Casper WY, USA
charlotte.gruner@caspercollege.edu
0009-0007-0522-8839

4th Mary Smith
Computer Science and Engineering
Hawai'i Pacific University
Honolulu HI, USA
mlsmith@hpu.edu
0000-0003-3912-0061

5th Xiaoyuan Suo
Computer and Information Sciences
Webster University
St. Louis MO, USA
xiaoyuansuo51@webster.edu
0000-0003-2473-0705

Abstract—

Parallel and distributed computing (PDC) has become the dominant computing model. Substantial research and advocacy efforts have been made to introduce PDC early in computing curricula. However, much of this research has relied on informal, ad-hoc measures of intervention efficacy. We address this gap by developing and validating an instrument designed to assess PDC knowledge of introductory computing students. This instrument is intended for use by researchers as a pre/post intervention assessment. We validate our instrument using Item Response Theory under the Rasch (IPL) model using data from multiple institutions. The instrument demonstrated a broad range of item difficulties, acceptable fit statistics, and evidence of unidimensionality. After recalibration, the improved instrument demonstrates strong reliability as a measure of foundational PDC knowledge. This instrument will help improve the rigor and quality of future PDC education research.

Index Terms—validated instrument, early computing education, CS1, parallel and distributed computing

I. INTRODUCTION

Multicore architecture has served as the standard hardware paradigm for the last twenty years. Substantial research and advocacy efforts have been made to include Parallel and Distributed Computing (PDC) early so that students consider it a natural computing paradigm rather than an advanced or rarely used concept [1].

Much of this research into various interventions has used ad-hoc measures to quantify effectiveness. In fact, a systematic review of literature found that 85% of PDC education papers were essentially experience reports which included instructor perspectives but little to no theoretical educational research

This material is based upon work supported by NSF OAC-2321015, OAC-2321020. We thank everyone at the Center for Parallel and Distributed Computing Curriculum Development and Educational Resources for their assistance.

results [2]. To address this situation and to allow comparability between interventions, it is important to have a *validated* instrument specifically designed to measure learning gains in early CS courses (CS0/1/2).

While validated surveys can provide valuable insights into students' dispositions, motivations, and perceptions of their learning, and are relatively easy to administer, they are subject to biases [3]. Student perceptions of learning do not always match test scores or other observations [4]–[6].

Instead, our goal is to create a shared, *validated* instrument to directly measure student learning. Rigorous, high-quality education research relies on instruments for which there is systematic empirical evidence that they accurately and reliably measure what they are designed to measure. Such instruments are “important for researchers to reduce experimental error due to flawed assessments and to allow for comparisons between different experiments” [7].

Decker and McGill [8] observe the need for formal evaluations using standardized and validated instruments while noting that researchers often fail to do so, instead creating their own or relying on ad-hoc measures. They systematically identified 47 such validated instruments in the context of computing, none of which include PDC topics. While there are validated instruments for computing education broadly [7], [9]–[13], there is a notable gap in validated instruments for PDC-specific learning outcomes. This current work seeks to fill this gap.

We build on earlier work [14] that reports on the development of a $I = 31$ item (multiple choice question exam) instrument to measure basic knowledge of PDC topics by students in introductory computing courses (CS0/1/2). The instrument is designed for use as a pre/post test to assess the effectiveness of PDC interventions in these courses and aligns

with curricula standards [15], [16].

We validate this instrument using Item Response Theory (IRT) [17], a widely used approach in educational testing and psychology. IRT characterizes each item based primarily on its difficulty and discrimination parameters and models the probability of a correct response as a function of a respondent's underlying ability. Discrimination captures how well an item differentiates between students of differing ability levels. Specifically, we use a Rasch (1PL) model [18] and report on the item fit, person ability, fitness of questions and provide post-hoc analyses. After analyzing our instrument, we recalibrate and report on the final validated instrument.

II. RELATED WORK

Research in introductory computer science education has traditionally relied on attitudinal surveys and self-report measures, which offer limited evidence of students' actual conceptual understanding; a gap that has become increasingly consequential as curricula incorporate PDC at earlier stages.

Concept inventories (CIs) address this gap by targeting documented misconceptions through carefully designed items. A recent review identified 47 validated instruments across computer science education [8], spanning digital logic [19], introductory programming [13], [20], dynamic programming [21], and others [22]. Yet despite this substantial body of validated assessments and standardized exams such as AP Computer Science A [9] and computational thinking assessments [12] not one of these 47 instruments targets PDC-specific conceptual understanding, leaving a conspicuous absence in an otherwise well-instrumented field.

This absence is increasingly difficult to justify. PDC has grown central to undergraduate education with the prevalence of multicore, distributed, and cloud computing, and the ACM/IEEE-CS Joint Task Force now identifies PDC as a core knowledge area [23]. Active learning, scaffolding, and unplugged activities have proven effective for introducing these concepts before students encounter parallel programming tools [24], [25], demonstrating that even introductory students can engage meaningfully with PDC ideas [26]–[28]. But as PDC instruction expands, instructors and researchers have no validated means of confirming whether this engagement translates into genuine conceptual understanding.

This gap has immediate and practical consequences. Existing assessment approaches (assignments, exams, and self-efficacy surveys) offer little insight into students' understanding of PDC. Without a psychometrically validated PDC instrument, researchers cannot rigorously evaluate instructional interventions, identify misconceptions at scale, or compare conceptual growth across institutions. As PDC continues to move into early courses, this absence represents not just a gap in the literature but a barrier to evidence-based reform. The present work develops and validates such an instrument.

III. ITEM RESPONSE THEORY ANALYSIS

Item Response Theory. We use IRT to validate our instrument. IRT validates an instrument by estimating each

question's (item's) *difficulty* and *fit*. An item's difficulty (β_i) reflects the ability level at which a respondent has a 50% chance of answering correctly. An item's fit indicates how well the observed response patterns conform to the model's expectations, such as whether students of a given ability respond as predicted. A well-designed instrument should include items with a range of difficulty levels in order to measure a wide spectrum of skill levels. We chose to use the Rasch 1PL (one-parameter logistic) model [18] for its simplicity and because our sample size is moderate. The Rasch model also has the advantage that item difficulty estimates are theoretically invariant across samples of respondents (up to linear transformation). Extended IRT models like the 2PL model, which adds a discrimination parameter, and the 3PL model, which further adds a guessing parameter, require larger sample sizes for stable parameter estimation than the 1PL [29].

Exam Content & Development. The goal of the PDC exam is to measure introductory computing students' (CS0/1/2) basic knowledge of parallel and distributed computing topics. The instrument is designed for use as a pre/post test to assess the effectiveness of PDC interventions in early computing courses. Items were written to be general, avoiding specific programming languages or frameworks, so the exam can be applied across various instructional contexts.

The exam¹ was developed by a team of five PDC education researchers from a diverse set of institutions. Initially, 76 questions were drafted, and the team conducted a thorough content review, resulting in a final set of $I = 31$ multiple-choice questions. All items except one have four response options with a single correct answer. The questions were created to check whether students can

- 1) define Parallel Computing (PC)
- 2) define basic terminology (threads, cores, etc.)
- 3) recognize the benefits (speed up) of PC
- 4) identify scenarios (data parallel, embarrassingly parallel) that may benefit from PC
- 5) explain the limitations of PC
- 6) define Distributed Computing (DC)
- 7) define Asynchronous Computing (AC)
- 8) distinguish between sequential, parallel, distributed, and asynchronous computing

Table I summarizes the topic/subtopic coverage. For full details on the review process and the alignment with ACM [16] and PDC2020 [15] curriculum standards, see [14].

Sample Data. After filtering for incomplete responses and students who opted out of data use, 270 responses were collected across 12 different courses at five institutions. Courses consisted of mostly CS1 and CS2 courses and one higher level systems course. Specific sample sizes can be found in Table II. These were selected to cover a wide range of student abilities, which improves the stability of item calibration and the interpretability of fit statistics. 13 responses were perfect

¹Available at <https://github.com/cbourke/ValidatedPDCEXam-public>, see also Artifact Description Appendix

TABLE I: PDC Exam Topics & Subtopics. Some questions are tagged with multiple topics/subtopics.

Topic/SubTopic	Count
Parallel Computing	15
Performance	3
Performance Limits	1
Distributed Computing	9
Asynchronous Computing	4
Event Handling	1
Concurrent Computing	1
Synchronization	1
Sequential Computing	2
PDC Vocabulary	3

TABLE II: Initial sample sizes by institution and course type.

Institution	Course	Sample Size
Casper	CS1/CS2	11
Knox	CS2	47
HPU	CS1	5
HPU	Capstone	11
UNL	CS2	71
UNL	CS2 Honors	25
UNL	Systems	79
UNL	Systems (grad)	3
Webster	CS2	18
Total		270

scores and were excluded from the analysis for a final sample size of $n = 257$.

In the following sections we analyze item difficulty, person ability, item fit, and perform several post-hoc analyses.

A. Item Difficulty

Overall performance was slightly negatively skewed, with a mean score of 24.27/31 (78.3%) and a median of 26/31. The distribution was bimodal, with peaks at 26 and 28, as shown in Figure 1.

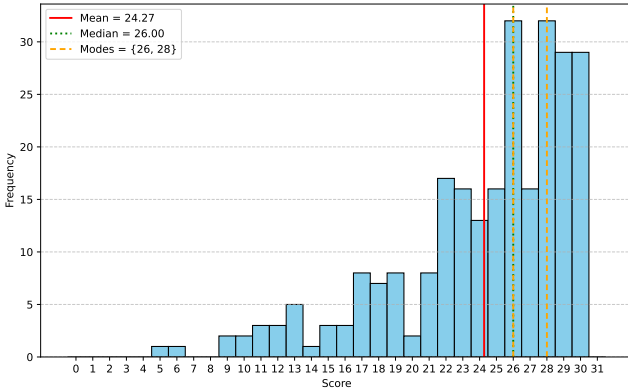


Fig. 1: Distribution of raw scores on the $I = 31$ -item instrument ($n = 257$). One point was awarded per correctly answered item. The histogram shows a slight negative skew.

Item difficulty (as well as person ability) is measured in logit (log-odds) units in the 1PL model. Specifically,

$$p(X_{ni} = 1) = \frac{\exp(\theta_n - \beta_i)}{1 + \exp(\theta_n - \beta_i)}$$

is the probability that person n responds correctly to item i , where θ_n is the person's ability and β_i is the item's difficulty.

$$\text{logit}(p) = \ln \left[\frac{p}{1-p} \right] = \theta_n - \beta_i$$

is the *logit* of p . Thus the logit maps a probability $p \in (0, 1)$ to a real number on $(-\infty, \infty)$, and in the 1PL model this logit is exactly the difference between person ability and item difficulty.

To estimate item difficulties, we used both Conditional Maximum Likelihood (CML) and Joint Maximum Likelihood Estimation (JMLE). CML conditions on each person's total score and provides unbiased estimates of item difficulties with respect to latent abilities, although it does not directly estimate person abilities. JMLE, in contrast, estimates *both* item difficulties (β_i) and person abilities (θ_n) iteratively via alternating Newton-Raphson updates. This approach is known to produce biased item estimates on short tests (few items), an instance of the incidental parameters problem [30], and yields undefined θ_n estimates for respondents with perfect or zero scores. To address this, we removed all such outliers and verified the resulting estimates. We found that, with only one exception (item 62, discussed later), all items had CML and JMLE values differing by less than 0.072 logits, below the expected standard error of item calibration [18]. This suggests that potential JMLE bias is negligible in this sample. For brevity, we report only JMLE values.

A full breakdown of item scores and Rasch difficulty estimates can be found in Table III. Item responses spanned a wide range. The most difficult item (62) was answered correctly by only 28.4% of students with a JMLE of 3.0925, while the easiest item (16) was answered correctly by 94.6% for a JMLE of -1.7203. With the exception of item 62, all JMLE values were in the range $[-1.72, 1.72]$ and so had an absolute value of less than two which makes them well-targeted [18]. Likewise the mean is essentially zero, meaning that the items are roughly targeted at the average ability sample. Standard errors are also reported with most of them having good precision ($\leq .2$ logits) with a mean of 0.192. A larger, but still acceptable error was observed for the "very easy" questions (those with a JMLE value ≤ -1). Our interpretation of difficulty (based on standard practices [31], [32]) for each question is also included.

We omit person abilities (θ_n) for every respondent, but report summary statistics (estimated using the previously discussed JMLE method). Person ability spans a broad range of latent skills, with a mean of $\theta = 1.85$ (SD = 1.30) and a range of -1.96 to 3.97 logits. Item difficulties ranged from -1.72 to 1.72 logits, excluding the one outlying item, indicating reasonable targeting in the low-to-middle ability range but limited coverage above $\theta \approx 1.72$. Mean standard error was 0.60, with higher SEs at the extremes; the elevated SEs among high-ability respondents likely reflect this lack of difficult items rather than distance from the average item difficulty per se, since standard error is governed by test information at each respondent's ability level. As the instrument is intended

TABLE III: Item Statistics. Accuracy includes the number ($n+$) and percentage (%+) respondents answered correctly. Difficulty includes JMLE estimates and standard error (SE) as well as our interpretation. Fit is reported including both infit and outfit each using both MNSQ and Z-values along with a final interpretation.

Item	Accuracy		Difficulty			Infit		Outfit		Fitness
	n+	Percent	JMLE	SE	Interpretation	MNSQ	ZSTD	MNSQ	ZSTD	
5	242	94.16	-1.637	0.284	Very Easy	1.077934	0.883442	1.483941	5.485846	Good Fit
6	228	88.72	-0.794	0.217	Moderately Easy	1.030960	0.350961	1.336232	3.811451	Good Fit
7	203	78.99	0.127	0.173	Slightly Hard	1.153973	1.745400	1.296022	3.355642	Good Fit
10	229	89.11	-0.841	0.220	Moderately Easy	1.095590	1.083586	0.954880	-0.511473	Good Fit
12	234	91.05	-1.102	0.238	Moderately Easy	0.978825	-0.240033	0.739384	-2.954285	Good Fit
13	182	70.82	0.696	0.157	Moderately Hard	1.043746	0.495897	1.083135	0.942404	Good Fit
16	243	94.55	-1.720	0.292	Very Easy	0.870633	-1.466476	0.446648	-6.272681	Good Fit
19	136	52.92	1.721	0.145	Very Hard	1.418172	4.740305	1.660690	7.489438	Underfit
20	185	71.98	0.621	0.159	Moderately Hard	1.063518	0.720026	1.023327	0.264429	Good Fit
26	175	68.09	0.866	0.154	Moderately Hard	1.153100	1.735512	1.088242	1.000292	Good Fit
27	212	82.49	-0.161	0.184	Slightly Easy	1.057257	0.649055	1.173931	1.971644	Good Fit
29	230	89.49	-0.890	0.223	Moderately Easy	0.924244	-0.858756	0.736531	-2.986632	Good Fit
30	226	87.94	-0.702	0.211	Moderately Easy	0.820214	-2.038011	0.554230	-5.053154	Marginal
35	175	68.09	0.866	0.154	Moderately Hard	0.888713	-1.261527	0.839057	-1.824413	Good Fit
36	241	93.77	-1.559	0.276	Very Easy	0.779156	-2.503443	0.319799	-7.710617	Marginal
37	217	84.44	-0.338	0.192	Slightly Easy	0.857351	-1.617043	0.665015	-3.797322	Good Fit
41	199	77.43	0.244	0.170	Slightly Hard	0.969635	-0.344212	0.879299	-1.368242	Good Fit
45	207	80.54	0.003	0.178	Slightly Hard	0.957891	-0.477338	0.981940	-0.204721	Good Fit
46	232	90.27	-0.993	0.230	Moderately Easy	0.817155	-2.072689	0.489515	-5.786743	Marginal
48	193	75.10	0.411	0.165	Slightly Hard	1.089378	1.013172	1.170582	1.933686	Good Fit
49	223	86.77	-0.573	0.204	Moderately Easy	1.039465	0.447362	1.181727	2.060013	Good Fit
50	154	59.92	1.339	0.147	Hard	1.004198	0.047592	1.131568	1.491424	Good Fit
54	202	78.60	0.157	0.172	Slightly Hard	0.898270	-1.153188	0.969058	-0.350753	Good Fit
57	214	83.27	-0.230	0.187	Slightly Easy	0.935546	-0.730642	0.716913	-3.209014	Good Fit
58	231	89.88	-0.941	0.226	Moderately Easy	0.871849	-1.452690	0.695196	-3.455197	Good Fit
62	73	28.40	3.093	0.156	Out of range	0.982006	-0.203978	1.556489	6.308236	Good Fit
68	210	81.71	-0.094	0.182	Slightly Easy	0.841138	-1.800830	0.608040	-4.443178	Good Fit
70	195	75.88	0.357	0.166	Slightly Hard	0.982411	-0.199382	0.822583	-2.011159	Good Fit
72	186	72.37	0.595	0.160	Moderately Hard	1.066303	0.751599	1.306966	3.479696	Good Fit
73	188	73.15	0.544	0.161	Moderately Hard	0.877160	-1.392489	0.761784	-2.700361	Good Fit
75	172	66.93	0.936	0.153	Moderately Hard	1.037579	0.425990	1.045584	0.516730	Good Fit

as a pre/post measure distinguishing lower- from higher-ability respondents, rather than discriminating finely among already high-ability examinees, this reduced precision at the upper extreme is not consequential for its intended use.

Figure 2 depicts a Wright Map, which visually aligns respondent abilities and item difficulties along the same latent continuum. Respondents are plotted on one side according to their estimated ability, while items are plotted on the other according to their difficulty. This visualization gives an intuitive assessment of targeting, revealing that items span the ability range of the sample and identifying items that are very easy or very hard relative to respondents. In this sample, person abilities ranged from -1.96 to 3.97 logits, and item difficulties ranged from -1.72 to 3.09 logits, indicating that most respondents are well-matched to the item difficulties.

Figure 3 shows the Item Characteristic Curves (ICCs) for all items, which represent the probability that a respondent with a given ability will answer an item correctly. Under the Rasch model, an item with a difficulty of -0.5 corresponds to a 50% probability of a correct response for a respondent with ability -0.5 logits. The curves are well distributed across the ability spectrum, with no excessive clustering, indicating that the set of items is appropriately targeted relative to the

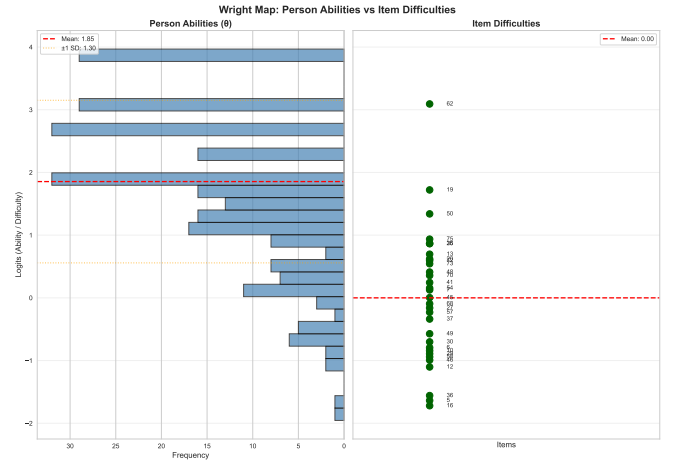


Fig. 2: Wright map (item-person map). Person ability (left) and item difficulty (right) on a shared logit scale.

range of respondent abilities. This suggests that the instrument provides meaningful measurement for both lower- and higher-ability respondents.

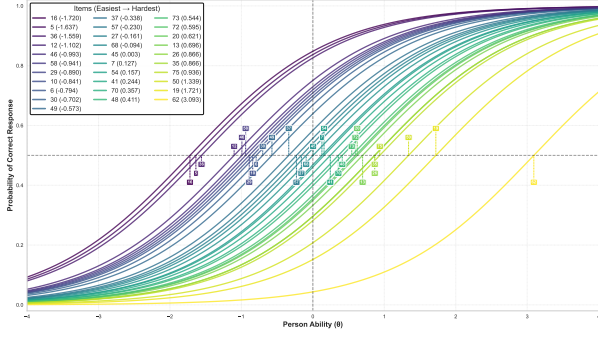


Fig. 3: Item Characteristic Curves. Each curve shows the modeled probability of a correct response as a function of person ability (logits). Item difficulty values are ordered easiest to hardest. Item difficulties span a good distribution across the ability continuum, $[-1.72, 1.72]$.

B. Item Fit

An item’s fit measures how well the observed responses conform to the Rasch model. We report both infit (information-weighted) and outfit (outlier-sensitive) statistics. Infit is most sensitive to unexpected responses on items near a person’s latent ability and less influenced by extreme responses, making it a reliable indicator of model misfit where it matters most for estimating ability. Outfit, in contrast, is sensitive to unexpected responses on items far from a person’s ability—for example, an easy item missed by a high-ability person or a hard item answered correctly by a low-ability person. It highlights misfit from extreme or atypical responses.

To assess item fit, we report both the standardized fit statistic (ZSTD) and the mean-square statistic (MNSQ). ZSTD reflects the statistical extremeness of an item’s misfit relative to the Rasch model and is dependent on sample size. It helps flag items that are statistically inconsistent with the model, even when the practical misfit is small. In contrast, MNSQ, the ratio of observed to expected residual variance, is independent of sample size and provides a stable measure of the magnitude of misfit for our sample [33]. We report both statistics because they serve complementary purposes: MNSQ quantifies how much an item misfits, while ZSTD indicates how confident we can be that the misfit is statistically significant.

Full statistics are reported in Table III. Items with infit or outfit ZSTD values within $|Z| \leq 2$ are generally considered to have good fit, with values closer to zero being ideal [18], [33]. For MNSQ, values between 0.7 and 1.3 are typically considered acceptable for productive measurement, with values closer to 1 being ideal [33].

Though we report both statistics for infit and outfit, when interpreting item fit, infit is emphasized for substantive decisions, as it reflects misfit on items near a person’s latent ability and is less influenced by extreme responses. Outfit, which is sensitive to unexpected responses on items far from a person’s ability, is primarily used to flag rare or extreme misfits that may not affect overall measurement quality. We first look at

MNSQ. Values below 0.7 are considered overfit (responses are more predictable than expected), while items with MNSQ values above 1.3 are considered underfit (responses are noisier than expected). For items with MNSQ within the acceptable range of 0.7–1.3, we examine the ZSTD, which accounts for sample size and indicates the statistical extremeness of the misfit. Items with $|Z| \leq 2$ are classified as having good fit, otherwise they are considered marginal, signaling statistically unusual response patterns despite acceptable practical fit. This dual criterion ensures that both the magnitude and statistical significance of misfit are considered.

Nearly all items (27/31) demonstrated good infit, with only three marginal items (30, 36, 46) and one underfit item (19). Regarding outfit, five items (5, 6, 19, 62, 72) exhibited underfit, although these are not depicted in Table III. Nineteen items showed good infit and outfit MNSQ values within the acceptable range (0.7–1.3) but had statistically significant ZSTD values ($|Z| > 2$). This is expected with our moderate sample size, as ZSTD can exaggerate statistical significance for outliers. Importantly, these items’ contributions to ability estimation are not compromised, as infit remains within acceptable limits, which primarily determines measurement reliability.

C. PCA of Residuals

The Rasch model assumes *unidimensionality*, meaning that one dominant latent trait—in this case, basic PDC knowledge—explains the item responses. To evaluate this assumption, we conducted a principal component analysis (PCA) of residuals. Residuals are the differences between observed responses and model-predicted responses, and PCA identifies whether these residuals exhibit patterns corresponding to secondary dimensions—additional latent traits not captured by the primary trait, such as general computing knowledge.

PCA of residuals revealed a first contrast with an eigenvalue of 2.325, accounting for 7.47% of the residual variance, suggesting a minor secondary dimension in the data. All subsequent contrasts had eigenvalues less than 2, indicating that the remaining residual variance is trivial. Overall, these results support the unidimensionality of the instrument. We further examined items with large loadings on the first contrast ($|loading| \geq 0.3$) and found that three items (16, 36, 58) contributed the most. All three items had positive loading values (0.350, 0.337, 0.406 respectively), pertained to parallel computing and were relatively easy, indicating that the minor secondary dimension likely reflects item topic clustering rather than a substantive threat to measurement.

D. Reliability

Item reliability reflects the internal consistency of the instrument, indicating how well the items collectively measure the intended construct. Cronbach’s α , ranging from 0 to 1, quantifies this consistency, with higher values indicating greater reliability. For our instrument, an alpha of 0.860 was observed, which falls in the “good” range, approaching the threshold for “excellent” [34]. This suggests that the items

provide a stable and coherent measure of the underlying latent trait.

E. Separation

Separation indices in Rasch analysis measure how well an instrument distinguishes respondents with different abilities and items with varying difficulty levels. Person separation indicates how precisely the test differentiates respondents along the latent trait, while item separation reflects how reliably the sample confirms the ordering of item difficulties. Together with their associated reliability coefficients, these statistics provide a comprehensive picture of the instrument's measurement precision and internal consistency.

For our instrument, the person separation reliability was 0.766, with a separation index of 1.809. This corresponds to approximately 2.7 ability strata, meaning the test can reliably distinguish roughly three levels of respondent proficiency. This demonstrates that the instrument provides sufficient precision to differentiate higher-, middle-, and lower-ability respondents along the latent trait continuum.

Item separation reliability was 0.965, with a separation index of 5.258, corresponding to approximately 7.3 strata of item difficulty. These results indicate that the sample size is adequate to confirm the relative ordering of item difficulties, providing highly stable measurements and confidence in which items are easier or harder for respondents.

IV. DISCUSSION & RECALIBRATION

The analysis demonstrates that our instrument is a reliable measure of introductory computing students' understanding of basic PDC concepts and provides evidence supporting its construct validity. Items cover a suitable range of difficulty and most exhibit good fit to the Rasch model, as indicated by infit and outfit statistics within acceptable ranges. PCA of residuals supports unidimensionality, indicating that the items primarily measure a single latent trait. Person and item reliability indices, along with separation measures, indicate that the instrument reliably distinguishes between students of differing ability levels and that the item hierarchy is stable. However, the analysis also identified several items that warrant further investigation. In the following section, we examine each item in detail to justify retention or removal and then recalibrate the instrument before rerunning the analysis.

Item 62 was identified as a difficulty outlier (JMLE = 3.093, exceeding the typical threshold of 2 logits). The item asked which response best exemplified distributed computing. The most common error (91/257) conflated distributed computing with team-based software development, distributing workload among developers. Other distractors were selected by 59/257 (problem decomposition) and 34/257 (parallel computing). This performance contrasts with other items addressing distributed computing, which were answered correctly by a larger proportion of respondents. Since the topic is otherwise well-represented and item 62 was substantially more difficult than the rest of the scale, we have removed it from the instrument.

With respect to item fit, item 19 was the sole underfitting item. Since all other items demonstrated good fit, this misfit is likely attributable to item wording or ambiguity rather than the model. The item asked respondents to choose the best example of asynchronous computing. The two most frequently chosen distractors conflated it with multitasking and distributed computing, suggesting that the underfit may have resulted from the conceptual similarity between multitasking and asynchrony. We have removed this item due to underfitting.

Three other items (30, 36, 46) were identified as having marginal fit. All three had acceptable MNSQ values, but ZSTD values below -2 indicated slight overfit. Items 30 and 46 were only slightly below this threshold (-2.03 and -2.07) and assess topics that are less frequently covered (event-driven and distributed computing, respectively); therefore, we retained both items. Item 36, however, exhibited greater overfit (-2.50) and assesses parallel computing, a topic already well-covered by other items. Consequently, we removed item 36 due to redundancy and overfit.

The PCA of residuals indicated that three items (16, 36, 58) had the largest loadings and contributed most to the first contrast's eigenvalue ($2.325 \geq 2$), suggesting potential local dependency or redundancy. Item 36 had already been removed, so we focus on items 16 and 58. These two items are conceptually redundant, both asking about the main objective (16) or goal (58) of parallel computing. To preserve unidimensionality and reduce redundancy, we have removed item 58, which had the larger loading value (0.406), and retained item 16 (0.350).

In summary, items 19, 36, 58, and 62 were removed, while items 16, 30, and 46 were retained, leaving a total of $I = 27$ items for recalibration. The remaining items provide adequate coverage for research purposes. After removing the flagged items, the analysis was rerun to assess the impact on model fit and scale properties. For brevity, only major changes are highlighted here; a summary of all changes is presented in Table IV.

After recalibration, key statistics remained largely stable. The effective sample size decreased slightly due to additional perfect scores being excluded (to avoid undefined JMLE values). However, it remains more than adequate for Rasch analysis. Measures of central tendency—including mean, median, and mode—were essentially unchanged, indicating stability of the person measures.

The range of item difficulties became slightly narrower and shifted marginally higher, with all items' JMLE values increasing slightly. The most notable improvement was in item fit: 26 of the 27 items now demonstrate good fit. Item 7, which previously exhibited good fit, now has marginal fit, with an acceptable MNSQ of 1.182 but a ZSTD of 2.005. This value slightly exceeds the conventional threshold of 2, indicating marginal misfit that is unlikely to meaningfully affect measurement quality.

Similarly, the PCA of residuals was updated following recalibration. The eigenvalue of the first contrast decreased to 1.939, below the commonly used threshold of 2, indicating no substantial secondary dimension and confirming that

TABLE IV: Statistics pre- and post-calibration.

Category	Statistic	Pre	Post
Item	Items (I)	31	27
	Sample Size (n)	257	243
	Mean score	24.27	21.31
	Mean Percentage	78.29	78.93
	Median	26	23
	Mode(s)	26, 28	26
Difficulty	Difficulty Range	[-1.72, 3.09]	[-1.65, 1.46]
	Mean (difficulty) SE	0.192	0.192
Ability	Ability Range	[-1.96, 3.97]	[-1.67, 3.53]
	Mean ability	1.854	1.828
	STDDEV	1.297	1.251
	Mean SE	0.595	0.632
Infit	Range	[0.779, 1.1418]	[0.834, 1.182]
	Mean	0.987	0.995
	STD DEV	0.127	0.099
Outfit	Range	[0.320, 1.661]	[0.477, 1.367]
	Mean	0.959	0.964
	STD DEV	0.326	0.246
PCA, first component	Eigenvalue	2.325	1.939
	Variance	7.47	7.15
Reliability	Cronbach's alpha	0.860	0.846
Person separation	Reliability	0.766	0.715
	Index	1.809	1.583
	Strata	2.74	2.44
Item separation	Reliability	0.965	0.943
	Index	5.258	4.069
	Strata	7.34	5.76

unidimensionality is now fully established.

Reliability metrics decreased slightly but remain high. The main effect of recalibration is a modest reduction in person separation. Item separation also decreased slightly but remains strong, corresponding to six distinguishable strata of item difficulty.

Based on the post-calibration improvements, the recalibrated 27-item instrument is recommended for use in PDC education research. We recommend administering it as a paired pre/post test instrument to measure learning gains when introducing PDC interventions in early computing courses as it provides a reliable and valid measure of students' understanding of foundational PDC concepts.

V. LIMITATIONS & FUTURE WORK

Several limitations should be noted when interpreting these results. Although the sample size was sufficient for stable Rasch calibration, additional data from broader institutional and curricular contexts, including different course levels, institutions, and student populations, would strengthen generalizability and support replication of the item parameter estimates.

Demographic-based differential item functioning (DIF) comparisons (e.g., gender) were not feasible given highly skewed subgroup sizes and incomplete background data. We did assess DIF (prior to calibration) with respect to students' self-reported prior PDC knowledge, split into higher (≥ 5 on a 7-point Likert scale, $n_{\text{some/high}} = 116$) and lower (< 5 , $n_{\text{low}} = 141$) groups, using the Rasch DIF-contrast (logit-difference) method [18] and the Mantel-Haenszel procedure [35]. Although several items showed non-trivial effect sizes none reached significance after Bonferroni correction ($\alpha = 0.05/31 \approx 0.0016$), likely reflecting insufficient power given

subgroup sizes rather than the absence of true differences. Because prior knowledge is closely aligned with the construct the instrument measures, even the observed effect sizes cannot be unambiguously attributed to item bias rather than genuine, exposure-related differences in ability. Measurement invariance with respect to student background therefore remains an open question for future study with larger, more demographically diverse samples.

More generally, several measurement limitations should be acknowledged. Although fit statistics and residual analyses supported the unidimensional structure of the instrument, unidimensionality should be interpreted in an essential rather than absolute sense, and minor secondary dimensions cannot be entirely excluded. As a multiple-choice instrument, scores may also contain construct-irrelevant variance due to reading demands, contextual interpretation, and distractor functioning (i.e., the degree to which incorrect options systematically attract responses). Finally, the present calibration does not include external validity evidence such as correlations with course performance or other measures of computing competence; without such evidence, it is not possible to confirm that scores reflect meaningful differences in PDC competence beyond performance on this specific item set. Future work should extend validation to additional populations, examine invariance across instructional conditions and pre/post intervention contexts, and assess the stability of item difficulty estimates across different instructional settings.

AI ACKNOWLEDGMENT

Claude Sonnet 5 (Anthropic, 2026) was used to generate Python glue code to utilize common Python libraries (`py-irt`, `numpy`, `pandas`, `matplotlib`, `scipy`) which processed data, ran statistical tests, and produced graphs. De-identified data was only processed locally in accordance with our IRB protocol. All code was manually verified and we stand by the accuracy of the work in this paper.

REFERENCES

- [1] Y. Ko, B. Burgstaller, and B. Scholz, "Parallel from the beginning: the case for multicore programming in the computer science undergraduate curriculum," in *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 415–420. [Online]. Available: <https://doi.org/10.1145/2445196.2445320>
- [2] Y. Sitsylitsyn, "Methods and tools for teaching parallel and distributed computing in universities: a systematic review of the literature," *SHS Web Conf.*, vol. 75, 2020. [Online]. Available: <https://doi.org/10.1051/shsconf/20207504017>
- [3] D. Tempelaar, B. Rienties, and Q. Nguyen, "Subjective data, objective data and the role of bias in predictive modelling: Lessons from a dispositional learning analytics application," *PLoS ONE*, vol. 15, 2020.
- [4] K. Craig, D. Hale, C. Grainger, and M. Stewart, "Evaluating metacognitive self-reports: Systematic reviews of the value of self-report in metacognitive research," *Metacognition and Learning*, vol. 15, no. 2, pp. 155–213, 2020. [Online]. Available: <https://doi.org/10.1007/s11409-020-09222-y>
- [5] K. S. Double, "Survey measures of metacognitive monitoring are often false," *Behavior Research Methods*, vol. 57, 2025. [Online]. Available: <https://doi.org/10.3758/s13428-025-02621-6>

- [6] J. Li and Y. Copur-Gencturk, "Perceptions versus performance: Assessing teacher learning in asynchronous online professional development," *Education and Information Technologies*, vol. 30, pp. 4751–4776, 2025.
- [7] M. C. Parker, M. Guzdial, and S. Engleman, "Replication, Validation, and Use of a Language Independent CS1 Knowledge Assessment," in *Proceedings of the 2016 ACM Conference on International Computing Education Research*. Melbourne VIC Australia: ACM, Aug. 2016, pp. 93–101. [Online]. Available: <https://dl.acm.org/doi/10.1145/2960310.2960316>
- [8] A. Decker and M. M. McGill, "A topical review of evaluation instruments for computing education," in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 558–564. [Online]. Available: <https://doi.org/10.1145/3287324.3287393>
- [9] College Board, "Advanced placement computer science a exam," 2023, offered since 1984; updated and revalidated annually. [Online]. Available: <https://apstudents.collegeboard.org/courses/ap-computer-science-a>
- [10] A. Elliott Tew, B. Dorn, and O. Schneider, "Toward a validated computing attitudes survey," in *Proceedings of the ninth annual international conference on International computing education research*. Auckland New Zealand: ACM, Sep. 2012, pp. 135–142. [Online]. Available: <https://dl.acm.org/doi/10.1145/2361276.2361303>
- [11] V. Ramalingam and S. Wiedenbeck, "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*, vol. 19, no. 4, pp. 367–381, 1998. [Online]. Available: <https://doi.org/10.2190/C670-Y3C8-LTJ1-CT3P>
- [12] M. S. Peteranetz, P. M. Morrow, and L.-K. Soh, "Development and Validation of the Computational Thinking Concepts and Skills Test," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. Portland OR USA: ACM, Feb. 2020, pp. 926–932. [Online]. Available: <https://dl.acm.org/doi/10.1145/3328778.3366813>
- [13] A. E. Tew and M. Guzdial, "The FCS1: a language independent assessment of CS1 knowledge," in *Proceedings of the 42nd ACM technical symposium on Computer science education*. Dallas TX USA: ACM, Mar. 2011, pp. 111–116. [Online]. Available: <https://dl.acm.org/doi/10.1145/1953163.1953200>
- [14] C. Bourke, D. P. Bunde, C. Gruner, M. Smith, and X. Suo, "Development and validation of a parallel and distributed computing exam for early computing students & research," in *2026 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, 2026, pp. 733–740. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/11652030>
- [15] S. K. Prasad, T. Estrada, S. Ghafoor, A. Gupta, K. Kant, C. Stunkel, A. Sussman, R. Vaidyanathan, C. Weems, K. Agrawal, M. Barnas, D. W. Brown, R. Bryant, D. P. Bunde, C. Busch, D. Deb, E. Freudenthal, J. Jaja, M. Parashar, C. Phillips, B. Robey, A. Rosenberg, E. Saule, and C. Shen, "NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing – Core Topics for Undergraduates, Version II-beta," Tech. Rep., Nov. 2020. [Online]. Available: <http://tcpp.cs.gsu.edu/curriculum/>
- [16] D. Lea, S. G. Aly, M. Oudshoorn, Q. Xiang, D. Grossman, V. Sarkar, M. Herlihy, S. Ghafoor, C. Weems, and S. Burckhardt, "Parallel and distributed computing (PDC) knowledge area – version gamma," <https://csed.acm.org/parallel-and-distributed-computing/>, Aug. 2023, version Gamma. [Online]. Available: <https://csed.acm.org/parallel-and-distributed-computing/>
- [17] S. E. Embretson and S. P. Reise, *Item Response Theory for Psychologists*, 1st ed. New York: Psychology Press, 2000. [Online]. Available: <https://doi.org/10.4324/9781410605269>
- [18] T. G. Bond and C. M. Fox, *Applying the Rasch Model: Fundamental Measurement in the Human Sciences*, 3rd ed. New York: Routledge, 2015. [Online]. Available: <https://doi.org/10.4324/9781315814698>
- [19] G. L. Herman, C. Zilles, and M. C. Loui, "A psychometric evaluation of the digital logic concept inventory," *Computer Science Education*, vol. 24, no. 4, pp. 277–303, Oct. 2014. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/08993408.2014.970781>
- [20] A. E. Tew and M. Guzdial, "Developing a validated assessment of fundamental CS1 concepts," in *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 97–101.
- [21] M. Ferland, V. N. Rao, A. Arora, D. van der Poel, M. Luu, R. Huynh, F. Reiber, S. Ossman, S. Poulsen, and M. Shindler, "Construction and preliminary validation of a dynamic programming concept inventory," *arXiv:2411.14655v1 [cs.DS]*, 2024. [Online]. Available: <https://arxiv.org/abs/2411.14655>
- [22] C. Taylor, D. Zingaro, L. Porter, K. C. Webb, C. B. Lee, and M. Clancy, "Computer science concept inventories: Past and future," *Computer Science Education*, vol. 24, no. 4, pp. 253–276, 2014.
- [23] A. N. Kumar, R. K. Raj, S. G. Aly, M. D. Anderson, B. A. Becker, R. L. Blumenthal, E. Eaton, S. L. Epstein, M. Goldweber, P. Jalote, D. Lea, M. Oudshoorn, M. Pias, S. Reiser, C. Servin, R. Simha, T. Winters, and Q. Xiang, *Computer Science Curricula 2023*. ACM Press and IEEE Computer Society Press and AAAI Press, 2024.
- [24] S. K. Ghafoor, D. W. Brown, M. Rogers, and T. Hines, "Unplugged activities to introduce parallel computing in introductory programming classes: An experience report," in *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '19)*. Association for Computing Machinery, 2019, pp. 158–159. [Online]. Available: <https://doi.org/10.1145/3304221.3325573>
- [25] T. Newhall, K. C. Webb, V. Chaganti, and A. Danner, "An introductory-level undergraduate CS course that introduces parallel computing," *Journal of Parallel and Distributed Computing*, vol. 199, p. 105044, May 2025. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0743731525000115>
- [26] S. A. Bogaerts, "One step at a time: Parallelism in an introductory programming course," *Journal of Parallel and Distributed Computing*, vol. 105, pp. 4–17, 2017. [Online]. Available: <https://doi.org/10.1016/j.jpdc.2016.12.024>
- [27] M. L. Smith, S. Srivastava, D. P. Bunde, A. Crockett, M. Gerten, P. Maher, J. Spacco, X. Suo, J. Wang, and M. Zhu, "A visual unplugged activity to introduce PDC," in *2025 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*. IEEE, 2025, pp. 658–665. [Online]. Available: <https://ieeexplore.ieee.org/document/11106112>
- [28] S. Srivastava, M. L. Smith, A. Ghimire, and S. Gao, "Assessing the integration of parallel and distributed computing in early undergraduate computer science curriculum using unplugged activities," in *2019 IEEE/ACM Workshop on Education for High-Performance Computing (EduHPC)*, Denver, CO, USA, 2019, pp. 17–24. [Online]. Available: https://tcpp.cs.gsu.edu/curriculum/?q=technical_program_eduhpc19
- [29] C. L. Hulin, R. I. Lissak, and F. Drasgow, "Recovery of two- and three-parameter logistic item characteristic curves: A Monte Carlo study," *Applied Psychological Measurement*, vol. 6, no. 3, pp. 249–260, 1982. [Online]. Available: <https://journals.sagepub.com/doi/10.1177/014662168200600301>
- [30] B. D. Wright, "The efficacy of unconditional maximum likelihood bias correction: Comment on Jansen, van den Wollenberg, and Wierda," *Applied Psychological Measurement*, vol. 12, no. 3, pp. 315–318, 1988. [Online]. Available: <https://journals.sagepub.com/doi/10.1177/014662168801200309>
- [31] B. D. Wright and G. N. Masters, *Rating Scale Analysis*. MESA Press, 1982.
- [32] R. K. Hambleton, H. Swaminathan, and H. J. Rogers, *Fundamentals of Item Response Theory*. Sage Publications, 1991.
- [33] J. M. Linacre, "What do infit and outfit, mean-square and standardized mean?" *Rasch Measurement Transactions*, vol. 16, no. 2, p. 878, 2002. [Online]. Available: <https://www.rasch.org/rmt/rmt162f.htm>
- [34] D. George and P. Mallery, *SPSS for Windows Step by Step: A Simple Guide and Reference, 11.0 Update*, 4th ed. Boston, MA: Allyn & Bacon, 2003, p. 231, rule-of-thumb for Cronbach's alpha interpretation.
- [35] P. W. Holland and D. T. Thayer, "Differential item performance and the Mantel-Haenszel procedure," in *Test Validity*, H. Wainer and H. I. Braun, Eds. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988, pp. 129–145.

Appendix: Artifact Description

Artifact Description (AD)

I. OVERVIEW OF CONTRIBUTIONS AND ARTIFACTS

A. Paper's Main Contributions

- C_1 An $I = 27$ item multiple choice exam that may be used as a pre/post test research instrument for interventions introducing parallel and distributed computing (PDC) in early computing curricula.
- C_2 Analysis, validation and calibration of the instrument using Item Response Theory, specifically the Rasch IPL model

B. Computational Artifacts

The primary and sole artifact is the multiple choice exam.

A_1 <https://github.com/cbourne/ValidatedPDCEXam-public>

Artifact ID	Contributions Supported	Related Paper Elements
A_1	C_1	N/A

II. ARTIFACT IDENTIFICATION

A. Computational Artifact A_1

Relation To Contributions

The multiple choice exam is intended as use as an instrument in parallel and distributed computing (PDC) educational research for early computing contexts. The instrument provides an objective means for measuring basic student knowledge of PDC concepts.

The exam is provided as a question bank in a canonical JSON format. The repository includes a Python parsing script that process the JSON file, randomizes responses (unless a question's answers have been marked ordered) and produces a markdown version. The script may be adapted to produce other formats for administration in a Learning Management System. A sample question can be found in Figure 1 with its JSON format in Figure 2.

- Which of the following best illustrates **Parallel Computing**
- Task A runs on computer X while Task B runs on computer Y, once both have completed, they report their results back to computer Z
 - Task B runs after Task A has been completed
 - Task A runs for a while, then pauses and allows Task B to execute, then resumes once Task B has completed
 - Task A runs on Processor X at the same time as Task B runs on Processor Y

Fig. 1: A sample question on parallel computing.

```

1 {
2   "id": 20,
3   "text": "Which of the following best illustrates
4   ↳ **Parallel Computing**?",
5   "topics": [
6     "parallel-computing"
7   ],
8   "is_ordered": false,
9   "choices": [
10    "Task A runs on computer X while Task B runs on
11    ↳ computer Y, once both have completed, they
12    ↳ report their results back to computer Z",
13    "Task B runs after Task A has been completed",
14    "Task A runs on Processor X at the same time as
15    ↳ Task B runs on Processor Y",
16    "Task A runs for a while, then pauses and
17    ↳ allows Task B to execute, then resumes once
18    ↳ Task B has completed"
19  ],
20  "correct_choice_index": 2
21 }

```

Fig. 2: Sample JSON question format.

Expected Results

Researchers introducing a PDC intervention in an introductory computing course are recommended to use this exam as a pre/post test instrument to determine if an intervention had a measurable impact on improving student understanding of PDC.

Expected Reproduction Time (in Minutes)

The time required to administer the exam and to collect data depends on the format with which it is administered. We recommend utilizing a Learning Management System to automate the process. Converting the exam to a format required by your LMS may take up to several hours and may be reduced by utilizing the scripts we've provided.

You can expect that each participant would take between 30 to 60 minutes to complete the exam.

Artifact Setup (incl. Inputs)

Hardware: Not applicable.

Software: Not applicable.

Datasets / Inputs: The paper itself contains the full analysis, validation and calibration. Data used to validate this instrument involved human subjects and was approved by each author's Institutional Review Board (IRB). IRB protocol precludes the public release of de-identified data for privacy.

Installation and Deployment: Not applicable.

Artifact Execution

Not applicable.

Artifact Analysis (incl. Outputs)

Not applicable.