The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #41712

The Justification Effect on Two-Tier Multiple-Choice Exams

Dr. Pablo Frank Bolton, Smith College

I am a Lecturer in the Computer Science department at Smith College. I received my PhD. from the George Washington University under the direction of Professor Rahul Simha. I currently teach a variety of undergraduate courses and have taught graduate courses in the past.

My research is currently focused on STEM, especially on the areas of identifying misconceptions, creating scalable and informative assessments, and in the use of active learning techniques such as learning-by-teaching, and peer learning.

In addition, I work on Human-Computer Interaction and how it might allow us to interact with virtual worlds and robots.

I enjoy collaborating with colleagues in other fields where I get to combine CS with Biology or Physics and play with their data.

Topics of interest include:

Flipped Classroom techniques to teach programming The benefits of games and puzzles in learning Construction of fair, scalable assessments Multimodal teaching with an emphasis on getting students to articulate their understanding 3D-Shape reconstruction and analysis The use of Embedded Systems and Machine Learning to automate (Biology) Laboratory tasks.

Liberty Rose Lehr, Smith College
Rahul Simha, The George Washington University
Michelle Lawson, Smith College

The Justification Effect on Two-Tier Multiple-Choice Exams

Abstract
Booming enrollment in computer science has raised the need for efficiently gradable assessments, among which are Multiple-Choice Question (MCQ) assessments. MCQs have drawbacks, among which are: random guessing, limited instructor insight into conceptual misunderstandings, student frustration in not being able to explain for partial credit. Recent years have seen several MCQ refinements that feature a two-tier structure that elicits justifications in addition to a correct-choice answer. These justifications can themselves be structured as choices to enable rapid grading. This paper introduces an additional refinement to the two-tier MCQ to acquire more information about a student's understanding by eliciting explanations for why the wrong options are wrong. In addition, this paper introduces a simple new metric, the Justification effect, or J-effect, that is easy to extract and apply in order to detect students that need help and identify questions that have design flaws. The use of this approach allows instructors to easily provide automated yet rich feedback to students and to pinpoint issues with test implementations, all while remaining easy to design, implement, answer and grade. The study explores the testing and implementation of three different trials over two semesters.

Introduction

Since its introduction in 1914, Multiple-Choice Question (MCQ) assessments have risen in popularity due to their efficient grading, duration-predictability, increase topic coverage, and resilience against instructor bias. Yet there are well-known disadvantages as well: students often guess and are frustrated that they cannot explain or receive partial credit, and for instructors, MCQs do not provide a detailed enough insight into student reasoning and potential misconceptions. Thus, recent efforts have focused on refining MCQ assessments with an additional *second tier* to each MCQ question that elicit student reasoning for their MCQ choices. The second tier either asks students to write justification text (Short-Response Questions or SRQs) or are themselves multiple-choice with a list of potential justifications to choose from (these are called Two-Tier MCQs or TT-MCQs [1]).

We propose **JMCQ** (Justified MCQ), a TT-MCQ assessment with an added twist to gain insight: students must additionally explain why wrong options in the MCQ are wrong by selecting (from choices) a short explanation. We reason that a single justification is also a single piece of data and perhaps a single point of failure (for the student) whereas multiple justification options for potential wrong answers might help build a more complete picture of a student's conceptual understanding. Because the two tiers provide two scores, a *correctness* score and a *justification* score, we seek to understand the degree to which one can *quantify* the level of misconceptions. To this effect, we introduce a scoring metric that we call the **Justification-effect**, or **J-effect** that, as we show, helps highlight misconceptions even for students who score well on correctness, and helps the instructor identify questions that were too hard or ill-posed. The study focused on the following research questions:

- **RQ1:** Does JMCQ expose a difference in correctness and justification grades?
- **RQ2:** Does the Justification-effect expose meaningful insights into student misconceptions and issues with question items?
- **RQ3:** Is the JMCQ protocol scalable?

The material objective of the process is to craft rich feedback for the students without causing additional burdens (for students and instructors). For that reason, we had two additional (secondary) research questions:

- Secondary question 1: Does JMCQ permit the creation of rich feedback?
- Secondary question 2: Does JMCQ help with test anxiety?

In brief, the data obtained answers these questions affirmatively. Note that JMCQ's grading can be fully automated and instantaneous when the justifications are themselves selected from among distractor-justifications. (In one of our trials we also used a text-input version for comparison.) Thus, JMCQ can be used to auto-generate a customized follow-up study plan for each student. Furthermore, because justifications are presented to students, it potentially removes some anxiety around whether a student's phrasing is sufficient.

What are effective use-case scenarios for JMCQs in computer science? While our study includes a theoretical course (algorithms) and introductory programming (CS1) we believe JMCQs can be applied broadly across the curriculum. It is perhaps best suited to formative assessments because summative exams are often high-stakes, lack feedback or remediation options, and face challenges with scoring, bias, reliability, and validity [2]. These types of assessments are also known to induce high levels of anxiety in students [3], a feature we seek to address with JMCQs. Encouraging students to explain, verify, and revise their answers allows formative assessments to alleviate anxiety and foster a growth mindset which has been shown to improve overall academic performance [4]. To design an effective formative assessment, several criteria must be met. The assessment should enable instructors and students to identify misconceptions through rich feedback and offer achievable opportunities to improve. Furthermore, to avoid excessively burdening educators, question construction should be straightforward (and questions reusable), and grading should be reliable and scalable. The challenge, then, is to design an assessment protocol that reveals gaps in struggling students' understanding and offers customized remediation, with a manageable instructor burden. This is the goal of our overall project.

In the upcoming section, we will review previous work and in section 3, we will provide an explanation of our research methods. In sections 4 and 5 we present and discuss our results. In section 6 we present our conclusions and future work.

Related Work

Background: summative vs. formative assessments. Traditional summative assessments are typically used at the end of a course to categorize a class into levels based on students' demonstrated skills and understanding. In contrast, formative assessments aim to diagnose and correct students' misconceptions during the course through timely and specific feedback. Adesope et al. discussed the benefits and range of summative and formative assessments [2]. Several researchers have found that formative assessments focusing on skills such as explaining, evaluating, analyzing, verifying, and revising answers, reduce anxiety and foster a growth mindset [2, 4–8]. In addition, empirical findings show that formative assessments encourage students to practice information retrieval more than other study methods (including summative assessments) [2]. Unlike summative assessments, formative assessments must provide timely feedback to enable real-time adjustments to the course material during the academic session [2, 9]. Even though low-stakes formative activities (as opposed to exams) have been shown to be very useful in promoting skill development [10], there are known advantages to using formative exams, such as the effect of motivating students to study and perform well [7, 8].

Background: benefits and disadvantages of MCQs. MCQ exams make grading convenient and improve overall recall by requiring test-takers to recall, compare, and analyze information about each plausible answer [11]. However, previous work has highlighted four notable limitations associated with MCQs: (1) constructing strong questions can be challenging; (2) they predominantly assess recall, a low Bloom level objective; (3) MCQs can yield inauthentic results due to unclear phrasing or guessing; (4) MCQs may prioritize test-taking skills above understanding [12]. In addition, MCQs may not fully reveal conceptual gaps [12–14].

In this paper, we focus on MCQs in spite of their disadvantages because instructors can use them to alleviate the pressures of grading in large classes, and because proposed two-tier refinements overcome some of their inherent disadvantages [15, 16]. With regard to question design, the main trial conducted in this study uses three-choice MCQs. Previous studies found evidence that questions with two or three misconception-based distractors are equally challenging [17, 18]. Furthermore, it was also noted that a total of three options reduced exam time, potentially improving score validity and reducing student anxiety [18].

Two-tiered questions. Studies show that students experience less test anxiety and prefer MCQs with options to explain answers further, as opposed to traditional MCQs [13, 19]. In the two-tiered approach, each MCQ is paired with an opportunity to justify, allowing a student some room for explaining their thought process. Tamir found that adding justification requirements could help educators diagnose misconceptions effectively [6, 15, 16]. The finding has led educators to design two-tier questions with various types of items [6, 14, 20, 21]. A two-tiered question can be deployed either free-response text (a Short-Response Question or SRQ) or by presenting an instructor-created justification (the correct justification) along with distractor-justifications. We believe the latter offers two strong advantages: (1) they are clearly easier to grade, and (2) analyzing and picking a justification may be more reliable than crafting them; in the work of Lee et al., even strong students found organizing ideas to generate original SR explanations more difficult than analyzing plausible MCQ explanations to select the correct answer [14]. We use the term TT-MCQ for a fully automatable choice-only two-tiered approach.

Some previous work has supported this approach as it enables instructors to assess students' ability to evaluate, analyze, and create explanations for answers while maintaining efficient and scalable grading [5, 22], with some studies showing that it reduces test anxiety [19, 21]. In computer science, previous research shows that using TT-MCQ-based learning activities improves overall retention of computer science concepts [10, 23]. Educators working on TT-MCQs recommend further studies to generalize the results [10, 14, 22, 23].

An enhancement: justifications for incorrect answers. Because a correct-answer justification is a single point-of-insight for a question, we sought to examine whether allowing students to explain why an incorrect option is wrong can provide additional insight into misconceptions (RQ2) and reduce test anxiety. While this notion has been studied in a short-response context and found beneficial [24], we believe our study is the first such TT-MCQ (with multiple justifications) in computer science. This mechanism has the direct advantage of reducing the grading effort (RQ3).

A second enhancement: a quantification of where justification diverges from mere correctness. Even if students selected tier-one's answer correctly, they could choose an incorrect justification, possibly leading to false understanding of the related concept [14, 22]. Consistently, students found tier-two more difficult, likely because understanding explanations demands more advanced knowledge of the material [24]. In this work, we focus on the insights obtained from the study of the differences between the tier one "correctness" questions and the tier-two "understanding" ones (RQ1). In particular, we propose and study the value of what we call the Justification Effect or J-Effect.

Methods

Two preliminary trials were conducted in the Fall of 2022 (F22), one at George Washington University and one at Smith College. The purpose of these trials was to assess the design of exams in terms of duration, the difficulty for students to answer and instructors to grade, and the variability of the answers provided as justifications. The lessons learned from these trials led to our final design for the Spring 2023 trial. All trials were IRB-approved with students signing a consent form to participate in the trial (which meant answering a pre/post survey). Otherwise, the grades from the normal coursework were anonymized and used for analysis. The overall consent rate was 91.6%, but for the preliminary trials the post survey response rate were so low (no matching post surveys in 80% of cases) that they had to be discarded. Only the main trial survey results are discussed.

Preliminary trials: A JMCQ protocol was used at George Washington University (GWU) for the course *Algorithms* ($N = 67$), attended mostly by second or third-year students. GWU is an R-1 University where students in the course were composed of 30% female and 70% male. A single midterm exam was administered in which students were provided a first tier MCQ with a SRQ justification (JSRQ) for each question (not for each choice). Students wrote a brief explanation that was later scored on a 5-point Likert scale. The trial used four-choice MCQ-JSRQ items, scored by obtaining the weighted average of both tiers, with the JSRQ accounting for two-thirds of each question's weight.

Another version of JMCQ was run at Smith College for Fall-2022 a CS1 course (in Python). Smith College is a historically women's liberal arts college. The course was graded with a Satisfactory/Unsatisfactory grade (S/U) but was run in such a way that scores were used throughout as in a normally graded course (a grade of $\geq 70\%$ corresponded to an S). The trial was run using a control group ($N = 19$) and two JMCQ groups ($N = 20$ and $N = 29$ respectively) taught by different instructors, the first of which also taught the control group. A midterm test was conducted on all sections, having the same questions but varying the item types. The control group was issued traditional MCQs while the other two had JMCQ with SRQ Justifications. Each three-choice MCQ had two distractors. The points for each MCQ were calculated as follows: correctly identifying a choice earned 3 points, and correctly identifying a distractor earned 1 point each (up to 5 points). For justifications, students had to indicate "this is correct" for the correct choice (1 point), and write two justifications for the two distractors (up to 2 points each). We conducted a surveys on student attitudes towards test structure and test-related anxiety.

Main trial: After studying the results from the preliminary trials (discussed in the Results section), we arrived at a simplified JMCQ design which was run, Spring of 2023 (S23), at Smith College for the same CS1 course. In this trial, a control section ($N = 30$) was issued traditional MCQ exams, and another section ($N = 30$) received our modified JMCQ exams. Both sections had the same instructor. Each three-choice MCQ had two distractors; a correctly identified choice gained 3 points, while correctly identified distractors were worth 1 point each. The following justifications were created:"this is correct" for the correct choice; a plausible but incorrect justification for the correct choice; a sound and a plausible justification for each distractor. In addition, two alternative answers were offered: "I don't know", and "other" (which required a comment). These two additional choices serve as an alternative for students that find the stated justifications confusing. In total, the justification tier was an eight-choice MCQ. This could be reduced to four by removing unsound justifications and alternative answers. However, the use of specific unsound justifications allows the detection of specific misconceptions while the reduced set only highlights a "confusion" between causes and effects for the given choices. All choices were afforded an optional comment box. The right justification for the correct answer, "this is correct" was worth 1 point, while the correct justification for each distractor was worth 2 points, all cumulative so that students were incentivized to get as many points as possible.

In this trial, the materials were presented in three segments, each followed by a 75-minute partial exam taken in class, accounting for only 15% of the final grade. Yet, students took the assessments seriously because of their cumulative significance. To encourage students to review and restudy their tests, we employed the following rule: the second and third partial tests contained a few questions that were slightly different versions of questions presented in the previous test. If students obtained a higher grade on these "second chance" questions, the grade would be averaged with the previous test's results. In this study we only report the raw grades with no points back. For this trial, we also conducted a survey on student attitudes towards test structure and test-related anxiety. It is important to note that the third exam was issued after students were given a very accurate estimate of their projected grades. This caused a change in their strategy for answering the test, where students who knew they were going to comfortably pass the course took future assessments less seriously. We call this the **S/U-effect** and discuss it in the next sections.

Question design for Algorithms: For the Algorithms course, questions followed a typical design for theory-course: given a puzzle involving an algorithm, students had to pick the correct answer from a four-choice MCQ. The justifications were Short-Response Questions (SRQs) where students were asked to explain their choice to satisfy the justification requirement.

Question design for introductory programming: For the case of the programming course, to design the JMCQs that had one justification per-choice, we followed the simple idea of reversing the focus of a programming question. Rather than asking about the results (effect) of a given program (cause), our JMCQs were designed by presenting the result of a computation (effect), and asking the student to identify the the program that produced it (cause). This simple notion allows the introduction of distractors based on misconceptions: the distractor would present a plausible “cause” that embodies a known misconception. *Questions designed like these enable educators to create plausible distractors based on known misconceptions*.

Designing a JMCQ (main trial): The process of designing an effective JMCQ that worked best was as follows (with example):

1. Pick a subject and *a few of the most common ways in which a student gets it wrong*. These will help define the justification distractors. Concepts inventories (with misconceptions) for Python, Java and other languages [25–27] are available.

   Example: A Python programming concept is the use of a single print statement to print a combination of multiple source strings. Students tend to mix-up syntactic elements from the multiple ways of doing this (comma-separated-printing, using a formatted string, and concatenation). Four common mistakes are:

   - Use of comma-separated printing but with explicit spaces between merged strings.
   - Mistake commas/spaces in the text for those in the print statement.
   - Use formatted printing but fail to use the `f-string` or `.format(...)` syntax.
   - Concatenate the component substrings but forget to include explicit spaces.

2. The prompt will be the desired effect of a correctly applied concept. One can also provide a template or structure as a given (which shortens the code for the multiple-choice options).

   Example: For our example, our prompt can be:

   `Given the following two lines of code:`

   ```
   str1 = "Hello"
   name = "Rose"
   ```

   `Which choice causes the following printout:`

   ```
   Hello, my name is Rose.
   ```

3. The three MCQ options are a) the correct code, b) a plausible program containing one of the noted common errors, and c) a plausible program containing another one of the noted common errors. Note that this uses only two of the most common errors. The others can be

used for justification distractors. Note that in the actual assessment, the options appear randomized.

Example: the following are the “cause” (choices) for MCQ.

a)

```
print(str1 + ", my name is " + name + ".")
```

b)

```
print("{str1}, my name is {name}.")
```

c)

```
print(str1,", my name is", name, ".")
```

4. To construct the 8 justification options, we include the following:

   (a) “this is correct” (to be used as the justification for the correct choice);

   (b) “I don’t know”;

   (c) “other” (which required a comment);

   (d) the (sound) justification for why the first distractor is incorrect;

   (e) the (sound) justification for why the second distractor is incorrect;

   (f-h) three plausible but unsound justifications. These three are crafted from the remaining common mistakes noted in step 1.

   It is important to note the following caveats:

   - The justifications should be written in plain language (as used in class).
   - The plausible but unsound justifications are invalid reasons of why an incorrect choice is wrong. As such, the important detail is that the reason of why the choice is wrong must not be expressed in the unsound justification. To craft these, you can express justifications for why other common mistakes are made (and that are not expressed in the MCQ options).
   - Only one of the justifications should match each distractor.

   For our example, The justifications could be the following (randomized in the actual assessment):

   i. “this is correct” (applies to choice a);

   ii. “I don’t know”;

   iii. “other”;

   iv. an `f` is missing before the print string to make it an f-string (applies to choice b);

   v. this statement adds unnecessary spaces (applies to choice c);

vi. this statement is missing necessary spaces (does not apply);

vii. this statement adds unnecessary commas (does not apply);

viii. this statement is missing necessary commas (does not apply).

Effort of crafting questions: In practice, each question was created from scratch in 10 to 15 minutes, or by starting from an existing MCQ question, in 5 to 10 minutes.

Grading insights: When grading, a mismatch might hint at a misconception. For example, distractor (v.) may be erroneously picked for option (a) if the student thinks all spaces in the statement affect spaces in the printout.

In summary, for each question, the student selects the correct answer and its justification, and then provides a justification for why the two distractors are not correct. All tests were implemented digitally and answered using computers. Table 1 presents a summary of the trials and Table 2 presents a summary of the exam formats for each trial (using the compact trial names). Table 2 shows the points awarded for each different combination of student responses (the bottom two rows in each table refer to the main trial).

Table 1: Trials.

Site	Period	Prof	Class	Treatment	Num Exams	MCQ Format	J-Type
GWU	F22	A	Algorithms	JMCQ	1	MCQ (4)	SRQ
Smith	F22	B	Programming	MCQ	1	MCQ (3)	NA
Smith	F22	B	Programming	JMCQ	1	MCQ (3)	SRQ
Smith	F22	C	Programming	JMCQ	1	MCQ (3)	SRQ
Smith	S23	C	Programming	MCQ	3	MCQ (3)	NA
Smith	S23	C	Programming	JMCQ	3	MCQ (3)	MCQ (8)

Target Measurements. In our analysis, we recorded the MCQ-only points (MCQ) and, when applicable, the justification-only points (J) of each one of our trials. The grade (G) for each exam was calculated as a weighted average of the MCQ correctness percentage, $MCQ_{\%}$, and the J-correctness percentage, $J_{\%}$. Note that only the GWU22-A-JMCQ trial used a different weight split than $50-50$. The **Justification Effect**, or J-effect in the main trial was defined as $G - MCQ_{\%}$. Note this is easily obtained as $(MCQ_{\%} + J_{\%})/2 - MCQ_{\%}$. This simplified formulation was chosen as an initial examination of direct results from each part of a JMCQ item, and is not yet focused on examining the difficulty of each tier (as in [24]).

Results

We present the overall exam grades as well as details relating to the differences in the correctness and justification grades. In the following tables and figures, the prefixes “G22”, “S22” and “S23”

Table 2: Formats.

Trial	Answer Points	Distractor Points (each)	J-Type	J-Points	Weighted Average
G22-A-JMCQ	1	0	SRQ	2 per question	$^{1}\!/_{3}MCQ$, $^{2}\!/_{3}J$
S22-B-MCQ	3	0	NA	NA	All MCQ
S22-B-JMCQ	3	1	SRQ	Correct:1 Distractor: 2 each.	$^{1}\!/_{2}MCQ$, $^{1}\!/_{2}J$
S22-C-JMCQ	3	1	SRQ	Correct:1 Distractor: 2 each.	$^{1}\!/_{2}MCQ$, $^{1}\!/_{2}J$
S23-C-MCQ	3	1	NA	NA	All MCQ
S23-C-JMCQ	3	1	MCQ (8)	Correct:1 Distractor: 2 each.	$^{1}\!/_{2}MCQ$, $^{1}\!/_{2}J$

refer to “Fall 2022 at GWU”, “Fall 2022 at Smith College”, and “Spring 2023 at Smith College” respectively. The three different instructors are denoted as “A”, “B” and “C”; the two types of treatments are “MCQ” and “JMCQ”. For the Spring 2023 semester at Smith College, three different partial exams were conducted, denoted by the addition of the partial exam number (“1”, “2”, or “3”).

MCQ-only vs Justification-only grades

The main research questions of this study focus on the differences between the MCQ results and the Justification results. The following are the differences observed, in the JMCQ trials, between the grades obtained solely on the MCQ scores and those obtained purely from the Justification scores. Later, the MCQ-only points are compared between treatments (MCQ vs. JMCQ).

Table 3 shows the MCQ-only and Justification-only grade results for each partial exam. The comparisons between them were computed using the Wilcoxon Rank-Sum test [28]. This test was selected because the general and tier grades were not normally distributed in general. There were no significant differences in the MCQ and Justification grades in all exams except for a moderate difference in the third partial exam for the Spring 23 introductory programming course (p = .002, r = .398). We believe this is, because most students already knew they had passed the course (or better) due to the S/U (Satisfactory/Unsatisfactory) grading of the course. We call this the **S/U-effect**. Note that grades are percentages.

Figure 1 shows the distributions for the grades. The overall trends show that Justification grades are slightly lower than those in MCQ, which is consistent with previous studies [24].

The bar plots in Figure 2 show the relation between each one of the exams that had both a MCQ control group and a JMCQ treatment. The MCQ sections can obtain either a correct (5-points) or incorrect (0-points) mark, which were scaled to 10 points to compare to the JMCQ questions. For the JMCQ sections a question can be marked correct or incorrect on the MCQ tier (up to 5 points) and they can also obtain full (5-points) or partial credit (between 1 and 4 points) on the Justification tier.

Table 3: MCQ vs Justification grades

Trial	Treat	Mean		Median		StDev	
		MCQ	J	MCQ	J	MCQ	J
G22-A	JMCQ	54.97	51.74	50.00	50.00	24.62	22.17
S22-B	MCQ	57.47	NA	58.30	NA	17.98	NA
S22-B	JMCQ	82.67	77.33	86.67	80.00	13.58	15.64
S22-C	JMCQ	81.38	77.24	86.67	81.67	14.62	17.65
S23-C-1	MCQ	80.08	NA	80.00	NA	14.63	NA
S23-C-2	MCQ	77.67	NA	85.00	NA	18.88	NA
S23-C-3	MCQ	68.67	NA	80.00	NA	22.70	NA
S23-C-1	JMCQ	81.33	75.33	80.00	76.25	12.52	16.34
S23-C-2	JMCQ	75.25	69.50	80.00	71.25	16.59	19.40
S23-C-3	JMCQ	75.86	55.17	80.00	55.00	24.13	24.80

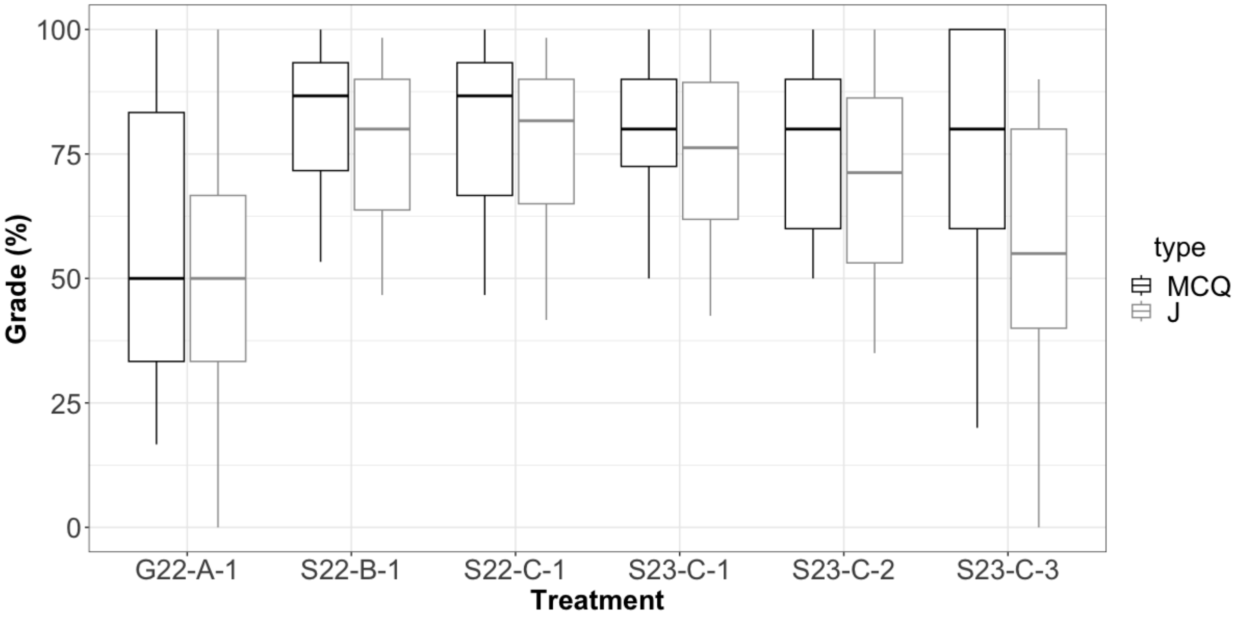


Figure 1: MCQ vs Justification Grades

To compare the proportions of correct answers between MCQ and JMCQ treatments (regardless of justifications), we used the two-proportion Z-test [28]. Two cases arose: one for the Smith College Fall 2022 introductory programming trial, in which the proportion of questions obtaining a correct answer in the JMCQ section was significantly higher ($MCQ = 62.2\%$, $JMCQ = 77.5\%$, $p = 0.001$). A different scenario occurred in the Smith College Spring 2023 introductory programming trial: no exams had significant differences between the proportion of correct answers in MCQ and JMCQ. The proportion of questions with correct answer (regardless of justification) for the MCQ (MCQ corr.) and JMCQ (J corr.) treatments, as well as the significance level of the difference between them (MCQ vs J corr.) can be seen in Table 4. However, when including the higher-resolution answers of the second tier, a radical change occurs: The Fall 2022 difference becomes non-significant and all Spring 2023 differences show that the Justification grades are significantly lower than the MCQ-only ones. The proportion of questions with full-credit answer (incorporating justifications) for JMCQ (J full), as well as the significance level of the difference between that section and the MCQ-correctness can be seen in Table 4 under the column MCQ vs J full.

While more work is needed to see which of these two cases is more likely in practice, our results indicate that there is a significant difference between an MCQ-correct answer and one that also considers the deeper (justified) understanding of the answer. This can be seen in the column "partial" of Table 4, where we indicate the percentage of questions that were marked "correct" in the MCQ tier did not receive "full" justification points. The average of the proportions of imperfect justifications from the set of questions marked correct was 31.38%. This means that, in the JMCQ sections, one third of the correct questions obtained less than full Justification credit (6 to 9 points). This is very similar to the proportion noted in the work of Lee et al. [14].

We can now respond to our first research question (**RQ1**): *a third of all responses marked "correct" demonstrated a lack of full understanding of the concept*. This is "hidden" in the MCQ-only view of the results and it can be used as a meaningful indicator of where to start in order to help students. In the following subsections, we look at different ways to obtain more in-depth insights with respect to individual students and question items.

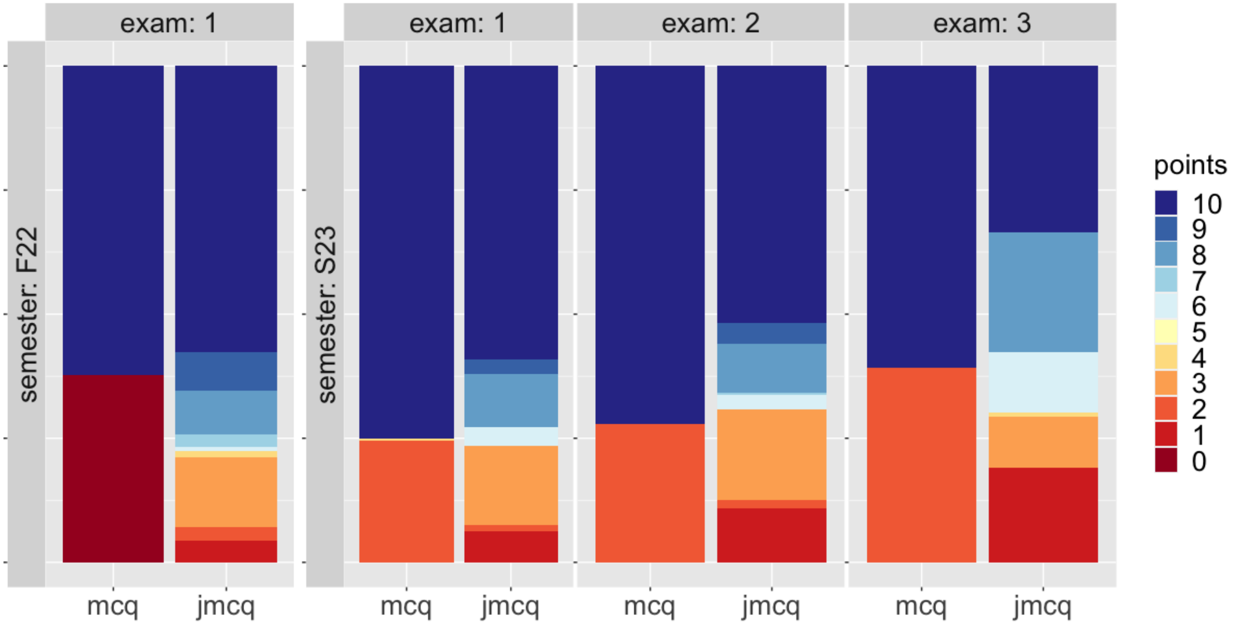


Figure 2: Proportions of points per question:
Proportion of questions receiving different number of points per exam.

J-effect for particular students and items

The J-effect is simply the difference between the overall exam grade and the one that would be obtained solely on correctness ($G - MCQ_{\%}$). Large values indicate a mismatch between correctness and justification precision. Negative values mean a lower Justification grade. We obtained the J-effect per question and by student in each JMCQ exam for the Spring 2023

Table 4: Correct vs Full credit proportions

sem	exam	MCQ corr.	J corr.	MCQ vs J corr.	J full	MCQ vs J full	partial
F22	1	.623	.776	p<.001	.578	p=.435	25.53%
S23	1	.750	.767	p=.749	.591	p<.001	22.82%
S23	2	.721	.692	p=.547	.517	p<.001	25.29%
S23	3	.608	.698	p=.188	.336	p<.001	51.86%

introductory programming trials. Figure 3 shows the average J-effect per student in every JMCQ exam. The values correspond to percent differences. All are negative because, in the calculation ($G - MCQ_{\%}$), the MCQ-only percent grade is, on average, higher than the combined JMCQ-percentage grade. This score can be used to notice issues at the question, student, and exam level. The Algorithms course (a hard class) had a wider range of values than the CS1 one. All CS1 courses, except the one for the third exam of Spring 2023 (S23-C3) were remarkably similar. In fact, the stark difference with that exam highlights one of the main benefits of this metric: it can exhibit issues with question quality, exam clarity, or other artifacts. In the case of S23-C3, the S/U-effect is clearly shown.

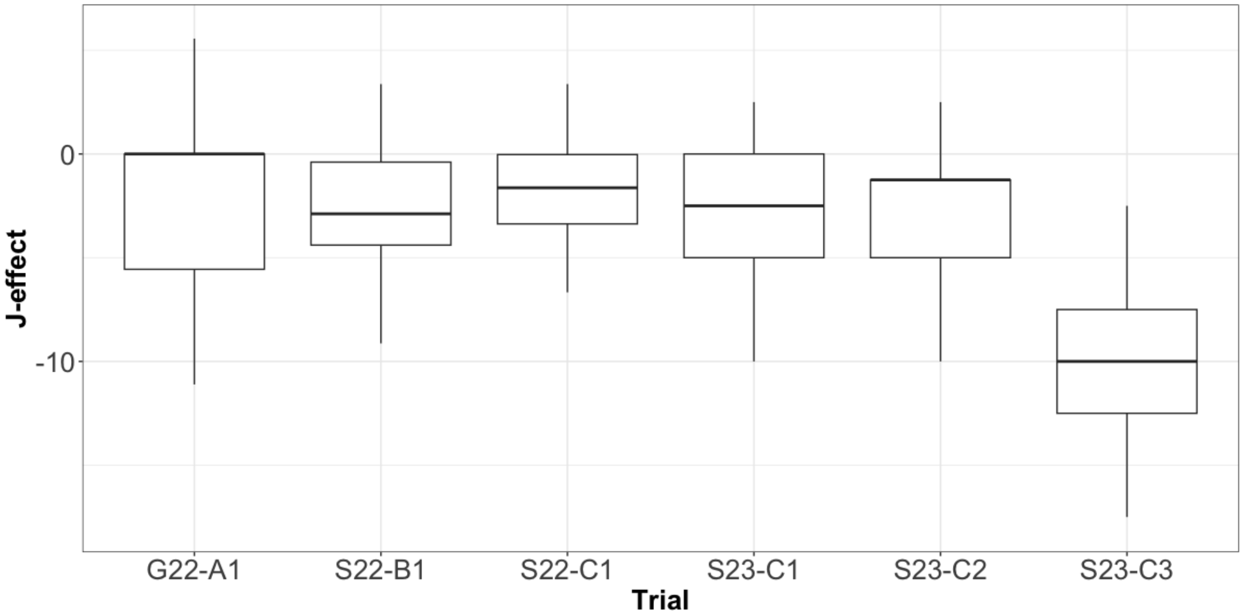


Figure 3: J-effect per Exam

J-effects per question: The J-effect average per question highlights the question's sensitivity to failure at the justification stage, denoting a possible link with a higher rate of misconceptions for the concept related to that question. In following with the findings of Lee et al. [14], we found that the mean discrimination index of the justification items (M =.46, SD =.19) was higher than that of the MCQ items (M = .39, SD = .22). However the difference was not significant. This is most probably due to the lower discrimination power of the justification items (since they are themselves an 8-option MCQ). While the discrimination indices for MCQ and J items were found to be strongly positively correlated (r(18) = .79, p = .032), there was no association between the J-effect and the discrimination indices. This indicates that, *for question analysis, the discrimination index and the justification effect may offer complementary views of a question's value* (**RQ2**).

J-effects per student: As mentioned before, the MCQ results "hide" a large proportion of imprecision in the area of Justification. An important fact to highlight is that the J-effect is not correlated with either the overall exam grade or the MCQ-only grade; perhaps more surprisingly, J-effect is also not correlated with the Justification-only grade; This points to *an effect that must be observed in the combination of MCQ and Justification answers and that can help identify and diagnose misconceptions, even in students with high grades* (**RQ2**). This is illustrated in the example case shown in Figure 4, which shows student results for the second exam in Spring 2023 (S23-C2). Exam grade (yellow) is superimposed on their MCQ grade (blue). Where the MCQ-grade is higher, blue peeks behind the gold grade (the grade yellow looks gold due to the blue color under it); where the MCQ was lower than the total grade, yellow can be seen peeking over the blue. It can bee seen that students have been arranged by a combination of their grade (high to low) and J-effect (low to high). It is clear that even students of moderate and high grades can have a relatively large J-effect and thus could benefit from additional help.

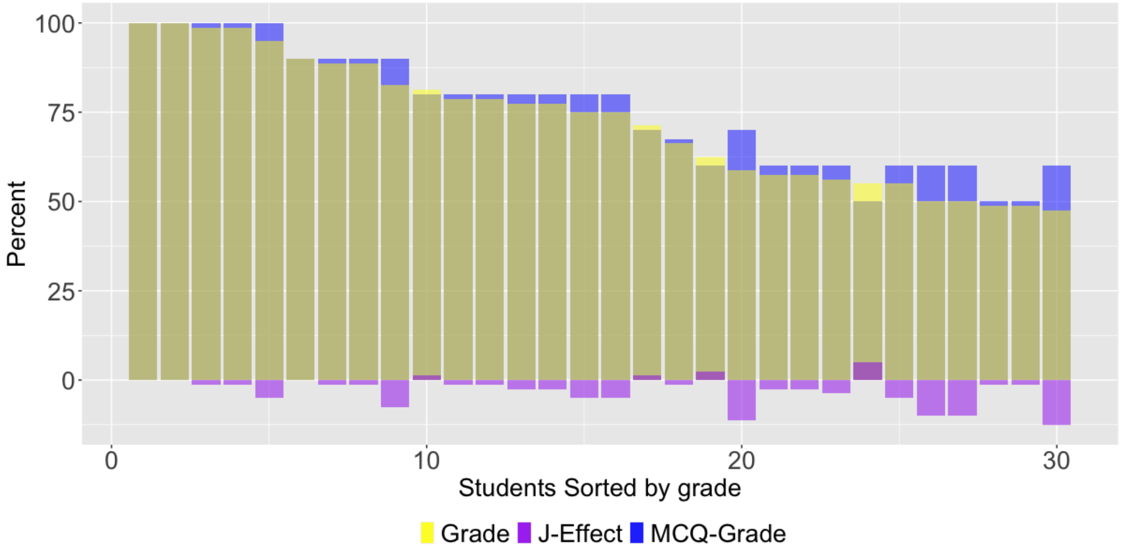


Figure 4: Combined Tier and J-effect per Student

Scalability and Feedback

The grading time for all MCQ treatments (across all trials with an MCQ section) was instantaneous. Minimal additional time was spent reading comments and awarding points back, amounting to less than 1 minute per student. Both Fall 2022 JMCQ treatments had an average grading time of 5 minutes per student. This amount of time is not too onerous for small sections but does not scale nicely. In the case of the GWU class ($N = 68$), the overall time was approximately 5 hours. For the Smith College Fall 2022 classes, the total time for the first JMCQ-section ($N = 20$) was approximately 1.5 hours; for the second JMCQ-section ($N = 29$), the time was approximately 2.5 hours. Fatigue and boredom have a known effect on grade reliability, even with the use of rubrics [29], and since rich and timely feedback is desired [2, 9] we considered that even this relatively short linear relationship (5 minutes per student) could be improved.

These results led us to modify the JMCQ treatments at Smith College during Spring 2023 and use a second tier multiple-choice question for the Justifications. After these modifications, the grading time dropped dramatically (instantaneous Justification grading) without a noticeable loss in the amount of information that could be gained about student reasoning and the possible misconceptions they might have. In all three partial exams for the JMCQ section, the grading was instantaneous. Similarly to the MCQ sections, the effort spent awarding additional points due to comments took fewer than a minute per student. *These results allow us to confirm that JMCQ is scalable* (**RQ3**).

In terms of feedback, reports were created for each student by making use of scripts that contrasted their answers with the expected ones. Of most use were the one on one office-hours sessions where specific per-student misconceptions were discussed and clarified. In short, *our approach allowed us to create individualized reports and feedback sessions to help students plan their next steps* (**Secondary question 1**).

Student Attitudes

The survey responses, shown in Figure 5 (questions are shown in shortened form), show no significant differences between the MCQ and JMCQ sections and overall attitudes were positive. Given the known benefits of MCQ over SRQ in terms of student anxiety, *we can infer that JMCQ features less anxiety than SRQ* (**Secondary question 2**).

MCQ vs JMCQ sections

The comparisons between treatments were computed using the Wilcoxon Rank-Sum test [28] and are shown in Table 5. There was a significantly higher average grade for both JMCQ treatments over the MCQ treatment of the Smith College F22 trial. This was not the case in the Smith College S23 trials. All trials for the introductory programming course taught at Smith College were similar except for the third partial exam of Spring 23, as will be further discussed in the next section, we believe this is due to the student's already knowing their projected final grades (S/U-effect).

Table 5: Trials Comparisons

Trial	Treat 1	Grade	Treat 2	Grade	P	r	size
Y22	B-MCQ	57.5	B-JMCQ	80.0	p< 001	.56	L
Y22	B-MCQ	57.5	C-JMCQ	79.3	p<.001	.51	L
Y22	B-JMCQ	80.0	C-JMCQ	79.3	p=1	0	S
Y23-1	C-MCQ	80.1	C-JMCQ	78.3	p=.666	.06	S
Y23-2	C-MCQ	77.7	C-JMCQ	72.4	p=.120	.2	S
Y23-3	C-MCQ	68.7	C-JMCQ	65.4	p=.783	.04	S

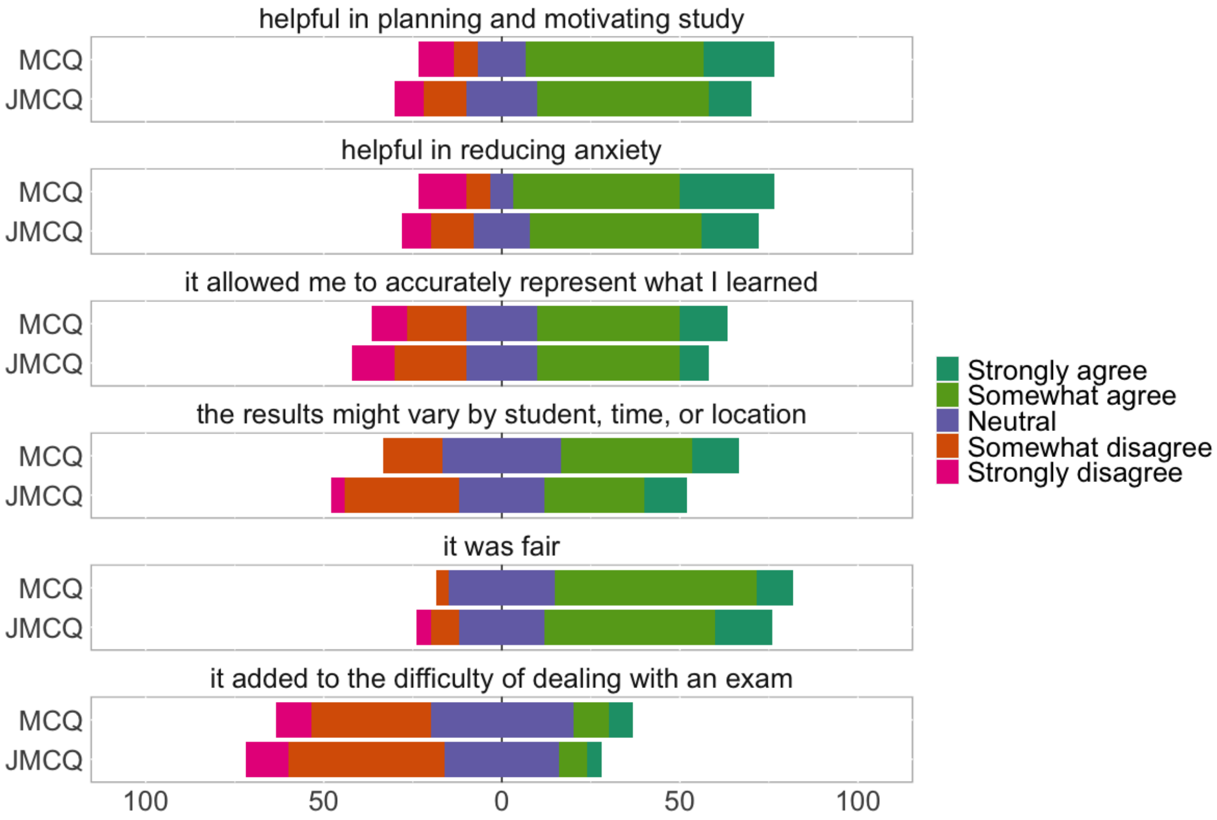


Figure 5: Survey responses

Discussion

This work presents a simple protocol to help design formative assessments that are easy to design and grade, and that can help reveal student misconceptions. The addition of a second tier of justifications for "why a distractor is incorrect" helped reveal gaps in understanding that would have not been highlighted on a traditional MCQ assessment. In particular, a third of all questions answered correctly in the MCQ part received less than full points in the justification part. This reveals two important things: 1) the MCQ-only grade returns, on average, overly optimistic results; and 2) the JMCQ version exposes an opportunity for follow-up with the student on specific misconceptions. This nuanced view into what students actually get right can be sen in Figure 2. While the MCQ proportions of correct vs incorrect are stark and inscrutable, the JMCQ proportions show a more precise distribution of "depth of understanding". It can be seen which "correct" questions are not fully understood (and why), and also hints to the misconceptions leading to an incorrect answer. One additional aspect to consider is that the results do not take into account the longer duration of JMCQ tests over the MCQ ones. A comparison with a longer MCQ test could allow an improved comparison with better control of student fatigue and the additional opportunities for mistakes.

We believe JMCQ can be used in any discipline where specific objective answers are linked to well defined causes. Disciplines in STEM tend to follow this general rule. However, if skill development is required (as in programming courses), the JMCQ assessments would aid in detecting and addressing misconceptions but would work best in conjunction with practical assignments (such as projects, exercises, or case studies).

An additional benefit is that questions can be saved in a bank and reused if they do not reveal large issues (such as having a large part of the class misidentify the correct justifications).

Limitations and Future Work: The main limitation is the need for further trials to replicate the findings, especially to distinguish if correctness grades vary, in the long run, between MCQ and JMCQ treatments. Any future comparison trials should take care to design exams of similar duration. Additionally, a follow-up trial could be used to refine the grading scheme (points awarded for each answer combination), and to determine the item and tier difficulties in JMCQ items (as demonstrated in [24]). In further trials, we plan to use the insights obtained from the J-effect to design student interviews based on their chosen justifications. These interviews would complement and enrich the feedback documents constructed for this study, and have the added benefit of providing an alternative way to corroborate the authenticity of student responses. In addition, the point-distributions, combined with the J-effect allows a view of responses that appear to be guesses (no Justification points). We would like to verify this with ad-hoc survey questions. Lastly, additional work needs to be performed to explore, in depth, the question of student anxiety and ways in which to further reduce it.

Conclusion

We carried out a series of trials to help dial in a scalable and informative test. The use of Two-Tier Multiple-Choice Questions allowed us to create an insightful exam whose grading is highly efficient. This frees up time to offer students review, office-hour sessions, and student-customized reports. The study also revealed a pervasive gap between correctness and conceptual clarity in MCQ tests through the use of the J-effect metric. This metric is very simple to obtain and understand, and can be used to highlight issues with individual question items, students, and even exams. One such case was the verification of the S/U-effect at Smith College in Spring 2023 (third partial exam). This effect was due to students knowing (almost certainly) if they would get a grade of “S”, and would invariably feel more prone to guess or speed through the questions. Our findings allowed the instructor to change when and how students receive grade projections for the course. Also, the instructor was able to create customized and immediate feedback reports (generated with simple scripts and containing information that highlighted the differences in Justifications as well as the most likely misconception).

In summation, this study allowed us to highlight the limitations of MCQ exams with respect to conceptual understanding and emphasized the importance of justifications and the utility of the J-effect as a complementary metric to discrimination indices and item grades.

Acknowledgements

This material is based upon work supported in part by the National Science Foundation under Grant DUE-2141772.

References

[1] D. F. Treagust, "Diagnostic assessment of students' science knowledge," *Learning science in the schools: Research reforming practice*, vol. 1, pp. 327–436, 1995.

[2] O. O. Adesope, D. A. Trevisan, and N. Sundararajan, "Rethinking the use of tests: A meta-analysis of practice testing," *Review of Educational Research*, vol. 87, no. 3, pp. 659–701, 2017.

[3] J. C. Cassady, J. Budenz-Anders, G. Pavlechko, and W. Mock, "The effects of internet-based formative and summative assessment on test anxiety, perceptions of threat, and achievement.," in *Proceedings of the Annual Meeting of the American Educational Research Association*, ERIC, 2001.

[4] Z. Yan, R. B. King, and J. Y. Haw, "Formative assessment, growth mindset, and achievement: examining their relations in the east and the west," *Assessment in Education: Principles, Policy & Practice*, vol. 28, no. 5-6, pp. 676–702, 2021.

[5] P. Rintayati, H. Lukitasari, and A. Syawaludin, "Development of two-tier multiple choice test to assess indonesian elementary students' higher-order thinking skills.," *International Journal of Instruction*, vol. 14, no. 1, pp. 555–566, 2021.

[6] P. Tamir, "Some issues related to the use of justifications to multiple-choice answers," *Journal of Biological Education*, vol. 23, no. 4, pp. 285–292, 1989.

[7] C. Penk, C. Pöhlmann, and A. Roppelt, "The role of test-taking motivation for students' performance in low-stakes assessments: An investigation of school-track-specific differences," *Large-scale Assessments in Education*, vol. 2, pp. 1–17, 2014.

[8] M. Leenknecht, L. Wijnia, M. Köhlen, L. Fryer, R. Rikers, and S. Loyens, "Formative assessment as practice: The role of students' motivation," *Assessment & Evaluation in Higher Education*, vol. 46, no. 2, pp. 236–255, 2021.

[9] N. E. Sudakova, T. N. Savina, A. R. Masalimova, M. N. Mikhaylovsky, L. G. Karandeeva, and S. P. Zhdanov, "Online formative assessment in higher education: bibliometric analysis," *Education Sciences*, vol. 12, no. 3, p. 209, 2022.

[10] T.-C. Yang, G.-J. Hwang, S. J. Yang, and G.-H. Hwang, "A two-tier test-based approach to improving students' computer-programming skills in a web-based learning environment," *Journal of Educational Technology & Society*, vol. 18, no. 1, pp. 198–210, 2015.

[11] J. L. Little, E. L. Bjork, R. A. Bjork, and G. Angello, "Multiple-choice tests exonerated, at least of some charges: Fostering test-induced learning and avoiding test-induced forgetting," *Psychological science*, vol. 23, no. 11, pp. 1337–1344, 2012.

[12] A. Hegde, N. Ghosh, and V. Kumar, "Multiple choice questions with justifications," in *2014 IEEE Sixth International Conference on Technology for Education*, pp. 176–177, IEEE, 2014.

[13] J. W. Fisher, "Multiple-choice: Choosing the best options for more effective and less frustrating law school testing," *Cap. UL Rev.*, vol. 37, p. 119, 2008.

[14] H.-S. Lee, O. L. Liu, and M. C. Linn, "Validating measurement of knowledge integration in science using multiple-choice and explanation items," *Applied Measurement in Education*, vol. 24, no. 2, pp. 115–136, 2011.

[15] P. Tamir, "An alternative approach to the construction of multiple choice test items," *Journal of Biological Education*, vol. 5, no. 6, pp. 305–307, 1971.

[16] P. Tamir, "Justifying the selection of answers in multiple choice items," *International Journal of Science Education*, vol. 12, no. 5, pp. 563–573, 1990.

[17] K. D. Royal and M. R. Stockdale, "The impact of 3-option responses to multiple-choice questions on guessing strategies and cut score determinations," *Journal of Advances in Medical Education & Professionalism*, vol. 5, no. 2, p. 84, 2017.

[18] S. D. Schneid, C. Armour, Y. S. Park, R. Yudkowsky, and G. Bordage, “Reducing the number of options on multiple-choice questions: response time, psychometrics and standard setting,” *Medical Education*, vol. 48, no. 10, pp. 1020–1027, 2014.

[19] A. F. Nield and M. G. Wintre, “Multiple-choice questions with an option to comment: Student attitudes and use,” *Teaching of psychology*, vol. 13, no. 4, pp. 196–199, 1986.

[20] A. Chandrasegaran, D. F. Treagust, and M. Mocerino, “The development of a two-tier multiple-choice diagnostic instrument for evaluating secondary school students’ ability to describe and explain chemical reactions using multiple levels of representation,” *Chemistry Education Research and Practice*, vol. 8, no. 3, pp. 293–307, 2007.

[21] F. Haslam and D. F. Treagust, “Diagnosing secondary students’ misconceptions of photosynthesis and respiration in plants using a two-tier multiple choice instrument,” *Journal of biological education*, vol. 21, no. 3, pp. 203–211, 1987.

[22] O. L. Liu, H.-S. Lee, and M. C. Linn, “An investigation of explanation multiple-choice items in science assessment,” *Educational Assessment*, vol. 16, no. 3, pp. 164–184, 2011.

[23] G.-J. Hwang, L.-H. Tung, and J.-W. Fang, “Promoting students’ programming logic and problem-solving awareness with precision feedback: a two-tier test-based online programming training approach,” *Journal of Educational Computing Research*, vol. 60, no. 8, pp. 1895–1917, 2023.

[24] G. W. Fulmer, H.-E. Chu, D. F. Treagust, and K. Neumann, “Is it harder to know or to reason? analyzing two-tier science assessment items using the rasch measurement model,” *Asia-Pacific Science Education*, vol. 1, no. 1, pp. 1–16, 2015.

[25] R. Caceffo, S. Wolfman, K. S. Booth, and R. Azevedo, “Developing a computer science concept inventory for introductory programming,” in *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pp. 364–369, 2016.

[26] G. Gama, R. Caceffo, R. Souza, R. Bennati, T. Aparecida, I. Garcia, and R. Azevedo, “An antipattern documentation about misconceptions related to an introductory programming course in python,” *Institute of Computing, University of Campinas, Tech. Rep. IC-18-19*, p. 106, 2018.

[27] R. Caceffo, P. Frank-Bolton, R. Souza, and R. Azevedo, “Identifying and validating java misconceptions toward a cs1 concept inventory,” in *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, pp. 23–29, 2019.

[28] R Core Team, *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023.

[29] N. Barnes and H. Fives, *Managing classroom assessment to enhance student learning*. Routledge, 2020.

Appendix 1: Question Creation

The following is a detailed example for question creation. The example assumes the existence of a traditional short-response (SR) question. Note that even though the subject matter is python programming, the high-level steps should apply to most subjects.

Picking a question

While there is a way to transform almost any question to a JMCQ-format, selecting a suitable starting point makes the process much easier. To this effect, the best questions are those that have a “cause and effect” relationship.

Two typical SR-questions are based on “Cause-and-effect” or “requirements-based crafting”. In the former, the objective is usually to describe the expected outcome for a given set of conditions. The latter is usually the reverse.

The JMCQ design works best when starting from a “requirements-based crafting” so the following example shows how to get to that format from a “Cause-and-effect” question:

Example SR question prompt (“Cause-and-effect”)

```
Prompt:  What is the objective of the following function?
```

```
def get_string():
  while True:
    text = input("Give me a string: ")
    if "@" not in text and len(text) >= 5:
      return text
```

Example (correct) SR answer: *The function returns a user-provided text only if it is longer than 4 characters and it contains no @ symbols.*

A potential issue for assessment here could be the phrasing used by the student, the absence of references to the correct ideas, or the inclusion of additional concepts.

The same question can be redesigned to follow a “requirements-based crafting” format (but this is not a required step for making a JMCQ).

Example SR question prompt (“Crafting”)

```
Prompt:

    Write a piece of code that does the following:
    It repeatedly asks the user to provide an input string until
    the string has the following characteristics:
    1. the string has 5 symbols or more
    2. the string does not have the symbol ’@’ inside
    If the string has the correct format, it returns the string
```

Example SR answer: (a typical student response looks like this)

```
def get_string():
    while True
        text = input("Give me a string: ")
        if @ not in text or len(text) >= 5:
            break
        return text
```

Errors in the SR answer

Syntax errors: Note that line 2 is missing a colon (`:`) at the end of the `while True` statement, and that line 4 is missing quotes around the `@` symbol.

Logical error: in line 4, the correct logical operator should be `and`.

Syntax or Logical error: in line 6, the indentation is incorrect (the `return` should be outside the `while` loop), which means that the function returns nothing.

While syntax errors might be attributed to typos or rushing (and thus easily addressed), logical errors point to deeper conceptual issues, so an answer showing both (and mixed) errors makes the assessment harder to determine, and feedback harder to craft.

A second, more insidious problem is that the structure of the question does not focus on particular misconceptions. For the case of this question, the student might make an error related to indentation errors while crafting loops, or related to logical operations while crafting conditionals.

Defining the question target and its misconceptions

We first define the target subject. In this case, we wish to determine if there are misconceptions related to the crafting of `while` loops, and we wish to exclude issues with conditionals.

The first step should be to determine the common misconceptions related to errors in the responses (Cause). These can be gathered from previous responses or from resources such as concept inventories on the subject.

For the example, common misconceptions for this type of question are:

1. Students fail to identify the relation between loop and function termination.
2. Students construct while loops that never end.
3. Students use a loop entry condition that is determined inside the loop.
4. Students confuse the effect of using `continue` or `break` inside a loop.

Defining the JMCQ correctness choices

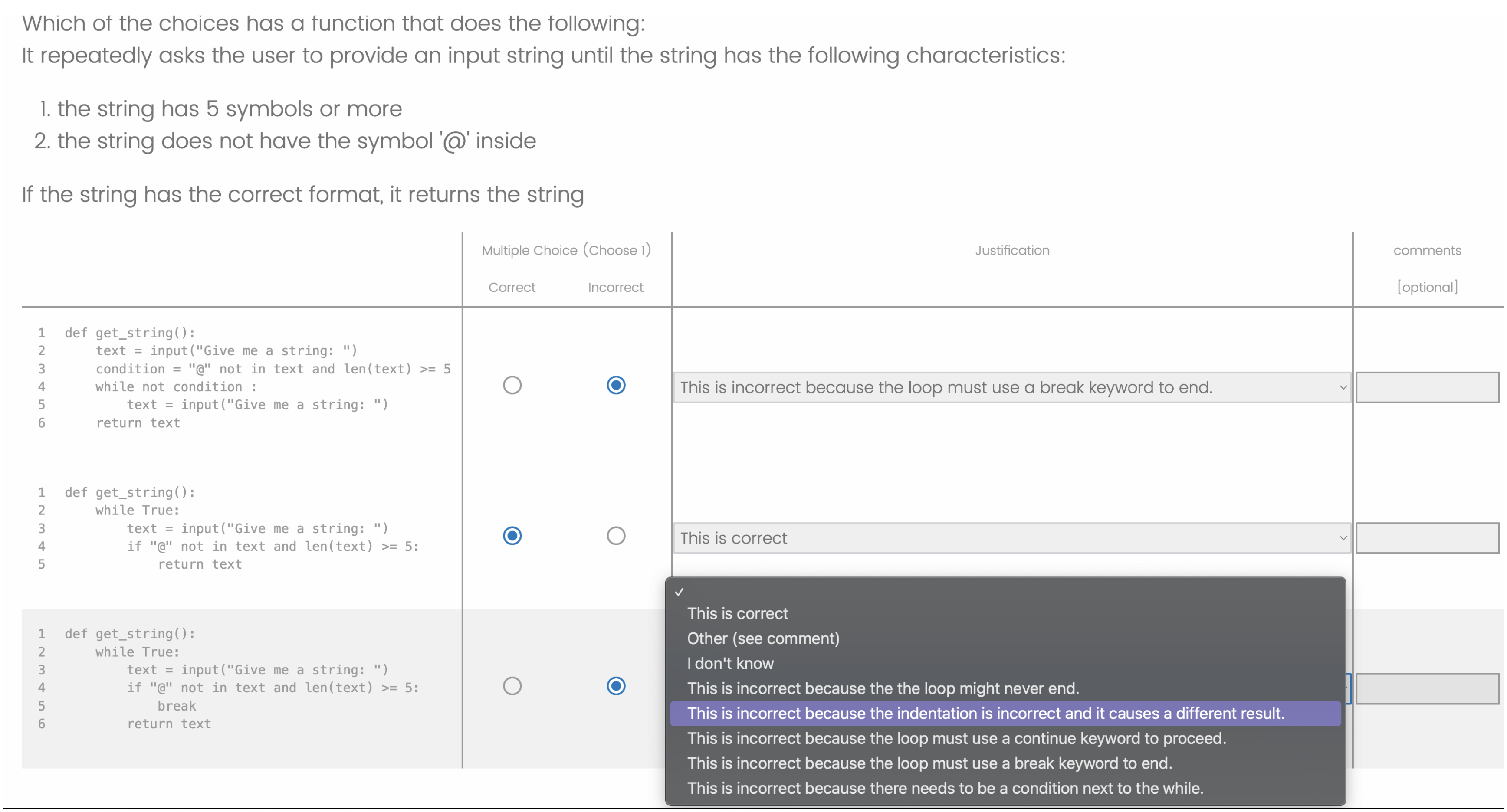
1. pick two misconceptions that we wish to detect (items 1 and 2 above).
2. Create answers that the students can pick: the correct answer as well as 2 distractors where the selected misconceptions are present.

.

Defining the JMCQ justification choices

1. Define the justification for the correct choice. This can be as simple as “this is correct”, or contain the actual reasoning behind the choice).
2. Pick the justifications for for why the chosen distractors are incorrect. These are simply the descriptions of the misconceptions.
3. Pick unsound justifications as distractors. These can be selected from the remaining (unused) misconceptions, or from other errors that sound plausible but do not apply.
4. Optionally, add options for when the student does not pick any justification (“I don’t know” and/or “Other”, with a comment).

The image shown below is a screen capture of a question used in an actual quiz (Using the Qualtrics survey system).



Which of the choices has a function that does the following:
It repeatedly asks the user to provide an input string until the string has the following characteristics:

1. the string has 5 symbols or more
2. the string does not have the symbol '@' inside

If the string has the correct format, it returns the string

```
def get_string():
    text = input("Give me a string: ")
    condition = "@" not in text and len(text) >= 5
    while not condition :
        text = input("Give me a string: ")
    return text
```

○ Correct ● Incorrect — This is incorrect because the loop must use a break keyword to end.

```
def get_string():
    while True:
        text = input("Give me a string: ")
        if "@" not in text and len(text) >= 5:
            return text
```

● Correct ○ Incorrect — This is correct

```
def get_string():
    while True:
        text = input("Give me a string: ")
        if "@" not in text and len(text) >= 5:
            break
        return text
```

○ Correct ● Incorrect

✓
This is correct
Other (see comment)
I don't know
This is incorrect because the the loop might never end.
This is incorrect because the indentation is incorrect and it causes a different result.
This is incorrect because the loop must use a continue keyword to proceed.
This is incorrect because the loop must use a break keyword to end.
This is incorrect because there needs to be a condition next to the while.

Figure 6: Example Question