Bob or Bot: Exploring ChatGPT’s Answers to University Computer Science Assessment

MIKE RICHARDS, KEVIN WAUGH, MARK SLAYemaker, MARrian PETRE, JOHN WOODTHORPe, and DANIEL GOOCH, School of Computing and Communications, The Open University, United Kingdom

Cheating has been a long-standing issue in university assessments. However, the release of ChatGPT and other free-to-use generative AI tools has provided a new and distinct method for cheating. Students can run many assessment questions through the tool and generate a superficially compelling answer, which may or may not be accurate. We ran a dual-anonymous “quality assurance” marking exercise across four end-of-module assessments across a distance university computer science (CS) curriculum. Each marker received five ChatGPT-generated scripts alongside ten student scripts. A total of 90 scripts were marked; every ChatGPT-generated script for the undergraduate modules received at least a passing grade (>40%), with all of the introductory module CS1 scripts receiving a distinction (>85%). None of the ChatGPT-taught postgraduate scripts received a passing grade (>50%). We also present the results of interviewing the markers and of running our sample scripts through a GPT-2 detector and the TurnItIn AI detector, which both identified every ChatGPT-generated script but differed in the number of false positives. As such, we contribute a baseline understanding of how the public release of generative AI is likely to significantly impact quality assurance processes. Our analysis demonstrates that in most cases, across a range of question formats, topics, and study levels, ChatGPT is at least capable of producing adequate answers for undergraduate assessment.

CCS Concepts: • Applied computing → Distance learning; • Computing methodologies → Artificial intelligence;

Additional Key Words and Phrases: ChatGPT, generative AI, cheating, quality assurance, university assessment’

ACM Reference format:

1 INTRODUCTION
In late November 2022, OpenAI opened public access to its ChatGPT program, an application built on top of the company’s GPT Large Language Model (LLM). ChatGPT showed unprecedented capabilities in generating textual responses to prompts and caused a sensation—not only

This project was funded entirely by The Open University.

This work is licensed under a Creative Commons Attribution-NoDerivs International 4.0 License.
© 2024 Copyright held by the owner/author(s).
1946-6226/2024/01-ART5
https://doi.org/10.1145/3633287

ACM Transactions on Computing Education, Vol. 24, No. 1, Article 5. Publication date: January 2024.
in academia and the fields of artificial intelligence and natural language understanding but also among the wider public. Within hours of the announcement, newspapers, magazines, and online sites were posting articles detailing their experiments with ChatGPT.

The enormous training set underpinning GPT-3 [23] allows ChatGPT to generate relatively large volumes of fluent text that—at least superficially—resembles that written by human authors. Such is the breadth of the data incorporated into the model that a wide range of potential problem domains have been proposed for ChatGPT—including summarizing complex documents, chatbots for customer inquiries, intelligent tutors, and in the role of a journalist. LLMs can produce large volumes of text that may be misleading, contain biases derived from inherent flaws in their training sets, or be entirely wrong. However, such is their apparent fluency that users and readers alike can be fooled into believing that responses are not only accurate but also the product of reasoning by an empathic intelligence.

Across the scientific and philosophy communities, there remains an active debate over whether LLMs understand language in any humanlike manner [35]. While Aguêra y Arcas has argued that “statistics do amount to understanding, in any falsifiable sense” [4], the authors tend to the view that LLMs lack any humanlike understanding of the meaning of their training data, and that they do not maintain any internal representation of the world. Researchers have described LLMs as “completely mindless” [25] and “stochastic parrots” [6].

The release of ChatGPT has not gone unnoticed in the educational community. In addition to discussion within academia, the wider media has also published many articles about the impact of ChatGPT on education, particularly its potentially destructive effect on assessment models [48]. ChatGPT provides a new approach for students to cheat by inputting assessment questions and receiving answers that can be passed off as a student’s work.

Academic integrity underpins institutional reputations. As Dick et al. point out [16], awarding academic qualifications to dishonest students has serious consequences—not only to the reputation of the institution itself but also to wider society: it damages the value placed on all academic qualifications; it damages the reputation of associated professions; and perhaps most seriously, it could endanger society as incompetent graduates enter employment and produce substandard or dangerous work. This prompted the first of our research questions:

- **RQ1:** What marks do ChatGPT-generated scripts on computer science (CS) final assessments receive from markers?

Given the importance of academic integrity to safeguarding the reputation of educational institutions and to wider society, the ability of existing marking processes to identify AI-generated scripts is considered in the second research question:

- **RQ2:** How well do experienced markers identify ChatGPT-generated scripts as containing “unusual or unexpected material” as part of their standard marking practices?

Our final research question considers whether newly developed software can reliably automate the process of detecting AI-generated material:

- **RQ3:** To what extent can current LLM detection tools distinguish between scripts produced by students and ChatGPT?

To empirically demonstrate the risk ChatGPT poses to academic integrity, we ran a “quality assurance” marking exercise using assessments from a range of computer science modules taken from our existing curriculum. Our results highlight that ChatGPT produces average answers that would pass most of the assessments used in this study. Markers could identify scripts containing unusual or unexpected material, although their formal flagging of scripts was extremely
selective. Automated methods of misconduct detection are currently insufficient to identify ChatGPT-generated scripts. We conclude by arguing that our analysis demonstrates the need to better explore which topic-specific assessment types are most resistant to generative AI, to ensure academic quality assurance while continuing to support students’ learning in an equitable manner.

Throughout this article, we focus on the version of ChatGPT based on GPT-3.5. This version of ChatGPT has no live Internet access and cannot provide information about recent events. Therefore, it has limited use as a tool for responding to current affairs; this is not the case with all LLMs (e.g., Google’s Bard). Since the work reported in this article was completed, further versions of GPT and ChatGPT have been released, which—alongside the creation of plugins to other services [55]—has improved the capabilities of the tool. Therefore, our results represent a snapshot of what a certain LLM tool could achieve at a certain time. However, we believe that as tools develop further capabilities and LLMs are incorporated into other software applications, the risks to academic integrity can only increase.

2 BACKGROUND LITERATURE

LLMs are developing at a much more rapid pace than the standard review cycle for most academic venues. As such, there is a lag in the published literature. In this section we first consider the broader context for academic misconduct and what motivates students to cheat. We then examine the impact of LLMs on assessment, before considering the literature on detecting LLM-enabled cheating. At the end of this section, we outline how our research questions help address some of the gaps in this fast-moving area.

2.1 Academic Misconduct

Cheating has been a threat to the integrity of education for as long as there have been examinations. In our own discipline of computer science, the conversation around cheating extends back several decades [17].

As Dick et al. point out; a simple definition of cheating is somewhat elusive; however, they proposed the following:

“A behavior may be defined as cheating if one of the two following questions can be answered in the positive:

- Does the behavior violate the rules that have been set for the assessment task?
- Does the behavior violate the accepted standard of student behavior at the institution? [16]

Furthermore, Dick et al. emphasize that institutions must make students aware of what is—and what is not—acceptable behavior, using systems such as student codes of conduct.

It is extremely difficult to determine the scale of cheating in academia due to the natural fear that students who admit to the practice may be penalized. Based on a randomized, privacy-preserving response, a study by Brunelle and Hott [11] surveyed students on an algorithms course (847 students), of whom 41.68% admitted cheating, and a CS theory course (181 students), in which half of the cohort was asked about cheating in coding (of whom 29.27% cheated) and half was asked about cheating in written assignments (31.31% cheated).

---

1 We used the then current version of ChatGPT, based on GPT-3.5, for this study. Unless otherwise stated, this is the version referred to by “ChatGPT.”

2 At the time of writing (March 2023), the “knowledge cut-off-date horizon” for ChatGPT is September 2021. Queries for events after this date receive a response such as: “I’m sorry, but as an artificial intelligence language model, I do not have access to real-time information or external data sources. Additionally, I do not have the capability to observe events or actions that occur in the physical world.”
Over a range of practices, the main influences on cheating in significant, large studies were found to be time pressure and concerns about failing, while the main countering influences were students’ desire to learn and to know what they have learned [39, 40]. Overemphasis on high grades may encourage students to engage in dishonest behaviors; Dyer et al. describe a need to develop greater academic integrity: “It [academia] needs to address the perception of higher education today as transactional in nature and of the need to get good grades as more important than the acquisition of knowledge” [18].

Some academic institutions have attempted novel assessment strategies to defeat—or at least deter—cheating; examples include generating unique exams for each student [44] or generating unique multiple-choice quizzes by selecting questions randomly from a large pool [14]. These assessment strategies prevent dissemination of answers between students sharing a common piece of assessment but are much less effective when LLMs can quickly and freely produce appropriate bespoke answers to individual students.

2.2 The Impact of LLMs on Assessment

It is highly likely that LLMs will become “just another tool” used in creative work; much as the electronic calculator, internet search engine, and computerized spellchecker have been incorporated into daily and professional lives as well as in education. Consequently, it is incumbent on educators to teach students what LLMs are, how they work, and how to make the best use of them [45]. LLMs potentially offer many educational benefits, including the potential to act as “informed” study companions, especially useful for students working alone and in distance education contexts; simplifying the development of personalized assessment; supporting creative writing; and engaging in the iterative development of software.

However, our focus is on the risks generative AI poses to academic conduct. Software such as ChatGPT is extremely convenient, easy to use, and provided at low (or zero) cost, making cheating using LLMs straightforward. Results from outside of computing highlight the ability for LLMs to produce answers to academic assessment. Huh used ChatGPT to answer questions from a parasitology exam taken by first-year medical students, comparing the answers to those from a corresponding cohort of students (n = 77) [28]. ChatGPT-generated answers scored 60.8% compared to a mean of 90.8% for the students (minimum score 89.0%, maximum score 93.7%).

Yeadon et al. outlined a study of ChatGPT’s performance in essay-based assessment for an undergraduate physics curriculum [54]. ChatGPT was used to generate 10 scripts for an exam composed of five questions requiring short-form (300-word) answers. It was found that the AI responded in a more discursive manner when given richer prompts including limiting word counts or including a known historical figure or event in the question. Not only was ChatGPT capable of quickly generating answers, but also these answers received an average mark of 71 ± 2%, with grades more tightly grouped around the average than in student populations. The researchers report that students performing in the lower third of their cohort would improve their grade if they relied on ChatGPT for their answers.

In contrast, an experiment with 18 undergraduate forensic science students found no evidence that ChatGPT helped students outperform peers with essay-writing tasks [5]. The cohort was split into two, with each group asked to answer the same essay question; one group had access to ChatGPT, and the other wrote essays using “traditional” resources under supervision to ensure that they could not use the chatbot. The control group outperformed those students who used ChatGPT, aligning with an earlier study by Fyfe in which students reported that incorporating ChatGPT outputs into essays was harder than writing the essay from scratch and did not represent a significant time saving [22].
2.2.1 The Performance of LLMs in Computing Academic Assessment. The relatively recent public availability of LLMs such as ChatGPT means that there is little in the published peer-reviewed literature that outlines the capabilities, shortcomings, benefits, and risks of the technology for educational purposes based on empirical investigation. In computing, much of the work focuses on software development. Outside of an educational setting, code-generating LLMs have shown a capacity to generate correct programs for common algorithms such as insertion sort and tree traversal [36]. However, user studies have shown that the productivity benefits for developers are mixed, making some tasks faster, but in other cases increasing the time required for debugging [51].

Jalil et al. took 31 questions from five chapters of a standard software testing textbook and generated three answers to each question using ChatGPT, also examining whether the answers were better if they were produced in a single-thread or shared context [30]. Their results indicate moderate performance, with 49.4% of the answers deemed correct if generated in a shared context (34.6% for separate threads), and 6.2% partially correct (7.4% for separate threads). Profiling 13 of the 31 incorrect answers, the authors identify a lack of knowledge or making incorrect assumptions as the cause of incorrect scripts.

Researchers at the University of Auckland focused on the capabilities of Codex, a new deep learning model trained on Python code from GitHub repositories, released by OpenAI, which can take English-language prompts and generate appropriate code. Taking 23 questions from two CS1 programming tests, Finnie-Ansley et al. found that Codex could correctly answer all but four of the questions using fewer than 10 responses, with nearly half of the questions (10) being solved successfully on the first attempt [19]. Overall, Codex’s score ranked 17th when compared to scores from 71 students. In a similar study, the authors used Codex to answer more complex CS2 advanced programming questions and again found that the system performed better than the average student [20].

Malinka et al. have tested various uses of ChatGPT—as a copy/paste tool, using an interpretation step, and as an assistant—across a range of different assessment types taken from four security courses [33]. Their findings established that ChatGPT produced answers in Czech that would have passed all courses except for one module, where simply copying and pasting its outputs led to scripts scoring just below the pass mark (50%).

Each of these studies focusing on the academic performance of an LLM [30, 19, 20, 33] used the research team or automated testing to assess the generated answers, rather than a marker who had no insight into the source of the answers. They each focus on the capabilities of the tool they are using (ChatGPT/Codex), rather than the broader process in which marks are awarded to scripts.

2.3 Detecting LLM-enabled Cheating

Significant effort has been invested in the detection of whether a given piece of text was written by a human or by an LLM such as ChatGPT. Chakraborty et al. argue that differentiating between human-originated and AI-generated text is possible but point out that as the capabilities of AI to generate humanlike text increases, the size of the sample needed for detection also increases [12]. Broadly, detection tools perform a statistical analysis of the text and provide a percentage likelihood of AI generation. Unlike plagiarism checkers, such as TurnItIn, which link a student submission to a pre-existing piece of text, these tools cannot demonstrate “proof” that a given piece of text has been generated by AI—only the probability that it may be machine generated, limiting their suitability in academic conduct cases.

There is a healthy body of literature of experiments using GPT-2 and other LLMs. Ippolito et al. noted that human discrimination of LLM outputs is generally lower than software detection and that, to disguise their origins, LLMs tend to add statistical anomalies that potentially allow
consistent detection of artificial scripts [29]. Rodriguez et al. note: “it is significantly harder to detect small amounts of generated text intermingled amongst real text” [43]. This is an obvious challenge, given that it is likely that most students using LLMs to cheat would use their outputs to supplement their own work rather than relying on an LLM to produce the entire submission. The ChatGPT-generated origins of text will almost inevitably be further disguised by students rewriting and manipulating the material or employing tools such as the grammar and style checkers built in to most modern word processors, or external grammar checking tools. Sophisticated cheaters could subvert the online tools used by universities for the detection of ChatGPT-generated texts to identify incriminating text in their submissions and make edits to change the suspicious content before submission.

Weber-Wulff et al. tested 12 publicly available tools and two commercial systems by examining 54 pieces of text generated by nine authors [52]. Each author created eight documents: two English-language documents written by themselves, two human-generated documents in a non-English language subsequently machine translated to English, two ChatGPT-generated documents, and two ChatGPT-generated documents that were then edited—one by a person, one paraphrased using the default settings for the online QuillBot AI service [56]. The detection tools were measured for their accuracy in determining the origin of each document.

TurnItIn was the most accurate of the detectors, with an overall score of nearly 80%. The detectors were 96% accurate in correctly recognizing human text, but accuracy fell sharply when presented with machine-translated texts of human origin and AI-generated text. The tools were only 42% accurate in the case of human-edited AI-generated text, a situation akin to students who wish to cheat but who make minor edits to an AI-generated document in order to disguise its origins (termed “patchwriting” by [27]). The most surprising result was that detectors were only 26% accurate in correctly determining the origin of machine-paraphrased AI-generated documents. These findings align with Anderson et al., who found that paraphrasing and editing could be used to avoid flagging by detection software [3].

Within computer science education, Orenstrakh et al. have presented an analysis of eight LLM detection tools [38]: GPT2 Output Detector, GLTR, CopyLeaks, GPTZero, AI Text Classifier, Originality, GPTKit, and CheckForAI.

Terms and conditions imposed by many LLM detection systems pose significant ethical and intellectual property rights issues as these systems claim rights over the submitted data. Of the eight systems explored by Orenstrakh et al., one no longer has an active online presence (AI Text Classifier), and two have no clear Terms of Use (GPT2 Output Detector and GLTR). The remaining five tools all contain clauses similar to: “By uploading content to our website, you grant to Originality.AI a perpetual, non-royalty-bearing, irrevocable, sublicensable, transferable, and worldwide license and right to retain and use an anonymized version of uploaded content to evaluate and improve our plagiarism and AI detection engines and services.”

Different countries and institutions will have different data and ethical protections placed around student data. At the Open University and under UK legislation, it is not possible to submit student scripts to any service that claims a right over the data without an institutional policy having been signed with the detection service organization.

Orenstrakh et al. used the eight tools to analyze 164 submissions (124 human submitted, 30 generated using ChatGPT, and 10 generated by ChatGPT and altered using QuillBot). The overall accuracy varied dramatically, with human data being identified between 54.39% and 99.12%, and ChatGPT data between 45% and 95%. False positives ranged between 0 and 52 [38]. The authors conclude “that while detectors manage to achieve a reasonable accuracy, they are still prone to flaws and can be challenging to interpret” and that they are not yet suitable for dealing with academic integrity.
2.4 Research Gap

This literature review highlights how cheating is a long-standing issue. ChatGPT poses specific issues in terms of convenience and cost, as well as the nature of its responses, rendering traditional plagiarism software inadequate. Much of the work in the area of computing has focused on software; further work is needed to both assess empirically the capabilities of ChatGPT to the broader computing curriculum and explore the risks it poses to academic quality assurance processes. We build on this work by ensuring that in our study design markers are unaware of the source of the scripts they are marking and by examining both the performance of ChatGPT-generated answers alongside exploring the broader marking process. This led to our three key research questions:

- RQ1: What marks do ChatGPT-generated scripts on CS final assessments receive from markers?
- RQ2: How well do experienced markers identify ChatGPT-generated scripts as containing “unusual or unexpected material” as part of their standard marking practices?
- RQ3: To what extent can current LLM detection tools distinguish between scripts produced by students and ChatGPT?

3 Methodology

To investigate these research questions, we designed a dual-anonymous study protocol focused on providing ChatGPT-generated and student scripts to markers who were asked to mark the work as part of a “quality assurance” study. The purpose of the study was masked from markers, posing it as a realistic marking scenario. While plagiarism detection forms an important part of academic integrity processes, it is not the primary focus of marking; therefore, our study simulates a typical marking process rather than acting as a “spot the AI” activity.

Before outlining our protocol, we provide some context as to how teaching and assessment occurs at the Open University.

3.1 How the Open University Teaches

All of the authors are based at the Open University, a large distance education university based in the UK. With more than 200,000 active students, it is the largest university in the UK awarding undergraduate and postgraduate degrees, as well as non-degree qualifications including diplomas, certificates, and continuing education units. Except for PhD students, the majority of Open University learners study part time using specially designed study materials intended for asynchronous distance learning developed by academic “module teams.” Module teams are formed of central academics who, in addition to developing the study materials, write the corresponding assessment material.

Open University modules are designed to be delivered at very large scale, with presentations of individual modules often exceeding 1,000 students. This is made possible through the role of “Associate Lecturers” (ALs), who each lead one or more tutor groups each containing up to 25 students. ALs have several responsibilities, including:

- Marking assessment following guidance from the module team
- Providing additional teaching through tutorials
- Acting as a point of contact for students for both study and pastoral purposes

Award-bearing qualifications from the Open University are assessed both during individual modules through one or more “Tutor Marked Assignments” (TMAs, broadly equivalent to coursework) and at the end of modules by an exam or an “End of Module Assignment” (EMA, similar to coursework but completed as the endpoint assessment).
The vast majority of student TMAs and EMAs are submitted electronically to the university’s assessment portal, where they can be retrieved for marking by ALs. As a response to the Covid-19 pandemic, the Open University moved from traditional invigilated face-to-face examinations to online open-book examinations. Final assessment markers are typically allocated 75 to 100 EMA/exam scripts, determined by the size of the module cohort.

With a few exceptions, for students studying on degree apprentice programs, a student’s EMA or exam script is marked by an AL teaching on their module who has not served in a tutoring role to that student. Other quality control measures include using automated plagiarism checks (TurnItIn and CopyCatch). Individual ALs can flag assessment scripts for review by module teams if they suspect content has been plagiarized, they have other academic conduct concerns, or they do not feel qualified to mark a script.

### 3.2 Module Selection

Four modules from across the undergraduate and taught postgraduate computing curricula were selected to explore ChatGPT’s ability to generate text at various levels of academic attainment and for different audiences. Distinct from much of the previous published work to date, we have focused on modules that do not predominantly teach software engineering or programming. The selected modules use a range of assessment methods and require students to demonstrate a range of academic skills. For the purposes of this study, we restricted our work to the final assessment, be that an exam or EMA. The selected modules are outlined in Table 1.

### 3.3 Selecting and Generating Scripts

The study used two forms of scripts: those previously submitted by students and those generated by ChatGPT. Each script was a complete exam or EMA submission containing answers to one or more questions (see Table 1). We had sufficient funds to pay each marker for the equivalent time to mark 15 scripts. Each marker received 10 student scripts and 5 ChatGPT-generated scripts—a sufficient number to determine if there was any underlying pattern in the AI-generated solutions to questions, but not so great as to make it obvious to markers that the scripts had not been generated by students.

#### 3.3.1 Student Scripts

The research team was given time-limited access to the university’s assessment archive. For each module, 10 student scripts were randomly selected from a cohort that commenced study in late 2021 (coded 21J/K below). Scripts were ordered by student identifier code (e.g., A1234567) and a random-number generator used to select scripts from the listing. This cohort was chosen not only because all grades had been finalized and any appeals processes completed but also because it was the last full cohort to finish studies before the public release of ChatGPT. We recognize that earlier, less accessible LLM tools could have been used to create these scripts but believe it unlikely (this is supported by the results in Section 5.2). While student scripts from older presentations were available, changes to module content and assessment approaches potentially meant that those documents were not representative of current teaching practice. This set of assessments was the first to have been written since the introduction of at-home examinations as a response to the Covid-19 pandemic.

#### 3.3.2 ChatGPT-generated Scripts

Five ChatGPT-generated scripts were generated for each module. In order to combine the outputs for individual questions into a script that resembled a student script, the following steps occurred: (1) generation and compilation of the ChatGPT answers, (2) removal of obvious ChatGPT artifacts from those answers, and (3) anonymization of all scripts presented to markers.
Table 1. Details on the Modules Used for the Study

<table>
<thead>
<tr>
<th>Module ID</th>
<th>Module Title</th>
<th>Level</th>
<th>Topics</th>
<th>Final Assessment</th>
<th>Format of Assessment Used in Study</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>Introduction to computing and information technology 2</td>
<td>Introductory</td>
<td>Python programming, computer architectures, cloud</td>
<td>End-of-module coursework</td>
<td>Seven questions, including short definition-based questions, a programming exercise, several short essays (up to 500 words), and a reflective component based on the student’s experience of the module.</td>
</tr>
<tr>
<td>CS2</td>
<td>Technologies in practice</td>
<td>Introductory</td>
<td>Samplers in computer networking, operating systems, and robotics</td>
<td>End-of-module assessment (EMA)</td>
<td>The EMA is designed to evaluate each of the samplers, with three short discursive questions and three longer essay questions requiring integration of knowledge and cited research. Essay templates are provided for the longer answers, of which the students answer any two of three. The final element is a reflective PDP (personal development planning) component.</td>
</tr>
<tr>
<td>CS3</td>
<td>Interaction design and the user experience</td>
<td>Advanced</td>
<td>Interaction design</td>
<td>Open-book exam</td>
<td>Fifteen short and one long question. Many of the short questions require students to briefly explain named concepts in interaction design in the context of specific scenarios. The long question concerns the development of a health-tracking application for senior citizens. Students create suitable questions that could be used in semi-structured interviews in the elicitation of requirements and to prototype an interface for the application answering a defined specification.</td>
</tr>
<tr>
<td>CS4</td>
<td>Information security</td>
<td>Postgraduate</td>
<td>Information security through the lens of the ISO27000 family of standards.</td>
<td>End-of-module assessment (EMA)</td>
<td>An extended report exploring a topic of the student’s choosing related to a case study developed during the module. As appropriate for a postgraduate module, the report is expected to be written to a high standard in “academic language” and to employ extensive referencing of module materials as well as external resources.</td>
</tr>
</tbody>
</table>

Due to staff capacity, five postgraduate students were paid £200 ($250) to produce the ChatGPT-generated scripts for a given undergraduate module, with the postgraduate CS4 scripts being generated by the research team. The instructions provided to the students noted that the chatbot should be prompted using each of the original assessment questions in a single ChatGPT thread. An extended dialogue between the user and chatbot was required for questions consisting of
Table 2. Mean Time Taken to Generate a Single ChatGPT-generated Script

<table>
<thead>
<tr>
<th>Module</th>
<th>Mean Time (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>23</td>
</tr>
<tr>
<td>CS2</td>
<td>15</td>
</tr>
<tr>
<td>CS3</td>
<td>32</td>
</tr>
<tr>
<td>CS4</td>
<td>25</td>
</tr>
</tbody>
</table>

multiple sections. If further detail was required, the chatbot would be requested to “continue” or repeat its explanation using alternative wording.

The chatbot’s outputs were copied into pre-prepared Microsoft Word template documents that had random styling (question numbers, font type, font size, titles) applied to reflect the diversity of styling found across student submissions. The students generating the ChatGPT-generated scripts were not allowed to add their own text, edit responses, or remove irrelevant material other than to remove any evidence the text had been generated by ChatGPT. The ChatGPT-generated scripts were then scrutinized by a member of the research team to ensure the removal of any evidence that they had been created using ChatGPT (e.g., the inclusion of text such as “As an AI language model…”).

This is the most naïve way of generating scripts through ChatGPT. We would anticipate cheating students to rephrase ChatGPT-generated material, adapt questions to account for image-based questions that couldn’t be answered, provide additional prompts to improve ChatGPT answers, and augment generated content with their own material. Our results therefore represent a baseline of minimal-effort cheating using ChatGPT, with naïve scripts more likely to be identified as suspicious by markers and detection software alike.

The students preparing the ChatGPT-generated scripts were asked to record how long it took to generate each script, as well as any problems they encountered. During this study, ChatGPT reported significant capacity issues resulting in slow performance and occasional periods of complete unavailability, meaning the times taken are at the upper end of the technology’s capability.

Table 2 shows the mean time taken to generate ChatGPT-generated scripts for the six modules. There was only one occasion on which script generation took more than an hour, but even this case would have allowed a student to complete the script within the time allocated for an examination (usually 2 or 3 hours). For modules using EMAs, students are allowed several weeks to develop their scripts, meaning that it is feasible to engage with ChatGPT over prolonged periods of engagement or over several sessions to complete a submission within the submission deadline.

It is not possible for us to release the prompts used to generate the scripts as these make use of university assessment material that we cannot publicly release. Our supplementary material provides an in-depth example of how each script was generated, as well as an example seven-page script generated from a publicly available sample exam. The final example script is indicative of what markers would have received.

3.4 Anonymizing Documents

Every effort was taken to anonymize student scripts. Scripts were anonymized and checked by two members of the research team to remove any identifying information (such as personal data or references to employers and sponsors) and then assigned new, unique identifiers.

Identifying metadata was removed from the student and ChatGPT-generated scripts, which were saved as Microsoft Word files. This not only removed author information but also reset the document creation and save dates, as well as erasing the document editing history. Had
this metadata been retained, it would have been relatively simple for markers to distinguish genuine work (written by students over extended periods during 2022) from AI-generated ChatGPT-generated scripts (created by the project team over a short period in early 2023 after the end of the normal module presentations).

At the Open University it is common for genuine student scripts to include minimal metadata and short editing times, due to students assembling final submissions from many shorter documents immediately before submission. There is no reason to believe that this does not happen elsewhere. Consequently, markers cannot rely on editing time metadata to identify suspicious scripts.

In the context of this study, removing the editing history from all documents does not impact the applicability of the results to typical marking processes and academic conduct across higher education.

### 3.5 TurnItIn Plagiarism Detection

Prior to marking, the 60 distinct scripts were passed through the TurnItIn plagiarism service, which identifies similarities with known sources of content. Module teams at the Open University often consider a TurnItIn score of 0.25 or above worthy of investigation for further evidence of plagiarism. One CS1 student script and four CS4 student scripts scored higher than 0.25. All ChatGPT-generated scripts produced TurnItIn scores lower than 0.21 (mean = 0.079), indicating they are composed of novel text rather than material sourced from elsewhere and that existing plagiarism tools cannot identify ChatGPT-generated material. The low TurnItIn scores for the ChatGPT scripts means they would not normally have been examined further for suspected plagiarism.

### 3.6 Markers

Markers were recruited from the existing AL community who typically mark EMA and exam scripts. CS1 and CS4 were marked by a single marker, with CS2 and CS3 scripts being independently double marked (a routine process performed by the Open University to ensure consistency between markers). Table 3 shows the marker ID for each module. The markers were paid for 2 days’ work, using the Open University day-rate equivalent for ALs. This resulted in 90 marked scripts, with 30 of the scripts double marked.

### 3.7 Marking

For each module, the 10 anonymized student scripts were mixed with the 5 ChatGPT-generated scripts and given filenames from 1 to 15. Ordering was randomized using a random number generator, with the exception that in any batch of scripts, one pair of ChatGPT-generated scripts was always given consecutive numbers (e.g., scripts 6 and 7). Script documents were copied to markers’ personal and confidential Microsoft Teams channels created for this study. Markers were then notified that the scripts were available for marking.

<table>
<thead>
<tr>
<th>Module</th>
<th>Marker ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>M1</td>
</tr>
<tr>
<td>CS2</td>
<td>M2, M3</td>
</tr>
<tr>
<td>CS3</td>
<td>M4, M5</td>
</tr>
<tr>
<td>CS4</td>
<td>M6</td>
</tr>
</tbody>
</table>
Markers were asked to provide feedback through a marking table (simulating an existing feedback system), where they also entered script marks. The feedback and script marks were essential for addressing RQ1. Markers were able to highlight suspected plagiarism or questionable content, recommend remarking, or identify problems with the script itself. Pre-existing marking guides developed by the module teams were provided. Markers were given up to 3 weeks to complete the task.

The markers were able to provide general feedback to the research team about the “quality” of an individual script. Guidance to markers about this feedback was deliberately vague, to avoid alerting them to the research being performed; markers were simply asked to flag “anything unusual or unexpected found in the scripts.” After completing marking, the markers uploaded the annotated scripts and feedback forms to their Microsoft Teams channels.

We concluded by inviting our markers to participate in a short semi-structured interview that was focused on addressing RQ2. Five of our six participants took part, with the interviews audio and video recorded. The interview script focused on individuals’ marking processes and how they typically detect and engage with academic conduct issues. After asking the markers what they thought the purpose of the marking exercise had been, further questions explored participants’ observations about student work, perception of ChatGPT, and their familiarity with the tool. During the interview, markers were asked to discuss any scripts that contained “unusual or unexpected material” regardless of whether they had previously been flagged during marking; this lowered the threshold for discussing suspicious scripts.

### 3.8 Study Perception

Table 4 summarizes the markers’ perceived purpose of the study and their experience with ChatGPT. It highlights a range of experience levels with ChatGPT and that we were broadly successful in masking the purpose of the study.

### 3.9 Analysis Techniques

A mix of descriptive statistics and graphing of the data was used to identify trends and patterns among the marks awarded. The interviews were audio recorded and transcribed. An inductive open coding approach was used to identify concepts and themes within the interview transcripts [10]. The transcripts were subjected to a line-by-line analysis by the first author, who had not interviewed any of the participants. Through this initial analysis, concepts were identified and labeled within the data. No codes existed prior to the analysis; they were created through constant comparison of the data and the application of labels to the text. This process was tempered by our interest in our research questions. These codes were subsequently categorized into unifying themes by the first author. These themes were then discussed with the interviewer, to ensure that

<table>
<thead>
<tr>
<th>Marker</th>
<th>Interview Length (minutes)</th>
<th>Perceived Study Purpose</th>
<th>Experience with ChatGPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>27</td>
<td>Identification of &quot;strange&quot; scripts</td>
<td>Not used, aware from the news</td>
</tr>
<tr>
<td>M2</td>
<td>41</td>
<td>Feedback on marking processes</td>
<td>Aware but not used</td>
</tr>
<tr>
<td>M3</td>
<td>42</td>
<td>Consistency of marking</td>
<td>Has tried with assessment material</td>
</tr>
<tr>
<td>M4</td>
<td>35</td>
<td>Consistency of marking</td>
<td>Aware but not used</td>
</tr>
<tr>
<td>M6</td>
<td>23</td>
<td>Chatbots</td>
<td>Aware but not used</td>
</tr>
</tbody>
</table>

M5 declined the invitation to be interviewed.
the developed themes corresponded to her interpretation of the data. No disagreements occurred during this discussion.

### 3.10 Data Protection and Ethical Considerations

The project team engaged fully with the Open University data protection and ethics policies, gaining appropriate permission from the institutional committees.

Since this study was solely for research purposes and was kept separate from formal module presentation and assessment procedures, the official grade for a student’s work would not be altered if a re-marked assignment was awarded a lower grade than on first marking. Likewise, disciplinary procedures would not be invoked if re-marking produced evidence of plagiarism.

Before starting the marking exercise, our markers completed a consent form. They confirmed that they understood that the interview was optional but that by confirming participation in the exercise, they would complete the allocated marking duties. If they opted to be interviewed, markers consented for it to be recorded and anonymous quotes to be used in publications. We allowed participants the right to withdraw their interview data within 2 weeks of the interview; none of the markers did so. The consent form also confirmed that the marking exercise data could be used for academic publications in an anonymized form and that the content and marking of scripts would be kept confidential.

We committed to informing the markers of the true purpose of the study, either at interview or, for those markers who did not want to be interviewed, by e-mail.

### 3.11 Academic Misconduct Software

In addition to the marking exercise, RQ3 motivated us to examine the ability of LLM detection software to distinguish between student-generated and ChatGPT-generated scripts. As noted in the literature review, many LLM detection tools claim a right over the data submitted. At the Open University and under UK legislation, it is not possible to submit student scripts to any such service without an institutional policy having been signed with the detection service organization.

Therefore, we used two tools compliant with our data protection regulations. The first was a previous generation of detection software that could be run locally without releasing any data to a third party (GPT-2: 1.5B [https://openai.com/research/gpt-2-1-5b-release](https://openai.com/research/gpt-2-1-5b-release)). We concede that this detection software is likely to be less successful at identifying ChatGPT outputs than a dedicated GPT-3 detector. In April 2023, the Open University chose not to opt out of the AI writing detection system offered as part of its existing TurnItIn contract, instead choosing not to use any comparison data from the AI tool as the basis for academic conduct cases. Given the Open University’s existing relationship with TurnItIn, it was possible to submit our sample set of 60 scripts for analysis. Details on the functionality of the TurnItIn software is available [57].

We suspect a script of being of AI origin if it scored more than 0.2. This threshold was selected as scripts identified by plagiarism software as being under a 0.2 match are rarely considered for further academic conduct action, with most module teams using a slightly higher threshold (0.25).

### 4 MARKING EXERCISE RESULTS

Driven by our first two research questions (RQ1: What marks do ChatGPT-generated scripts on CS final assessments receive from markers? and RQ2: How well do experienced markers identify ChatGPT-generated scripts as containing “unusual or unexpected material” as part of their standard marking practices?), our results are presented in four subsections. The first subsection covers the performance of the ChatGPT-generated scripts in the dual-anonymous marking by ALs, while
### Table 5. Final Percentage Scores for ChatGPT-generated and Student Scripts

<table>
<thead>
<tr>
<th>Script type</th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
<th>M5</th>
<th>M6</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChatGPT generated</td>
<td>80</td>
<td>67</td>
<td>54</td>
<td>63</td>
<td>51</td>
<td>13</td>
</tr>
<tr>
<td>ChatGPT generated</td>
<td>83</td>
<td>69</td>
<td>65</td>
<td>66</td>
<td>50</td>
<td>21</td>
</tr>
<tr>
<td>ChatGPT generated</td>
<td>84</td>
<td>60</td>
<td>66</td>
<td>66</td>
<td>44</td>
<td>36</td>
</tr>
<tr>
<td>ChatGPT generated</td>
<td>86</td>
<td>67</td>
<td>71</td>
<td>67</td>
<td>45</td>
<td>41</td>
</tr>
<tr>
<td>ChatGPT generated</td>
<td>86</td>
<td>52</td>
<td>55</td>
<td>66</td>
<td>46</td>
<td>16</td>
</tr>
<tr>
<td>Student</td>
<td>52</td>
<td>97</td>
<td>85</td>
<td>71</td>
<td>54</td>
<td>88</td>
</tr>
<tr>
<td>Student</td>
<td>91</td>
<td>84</td>
<td>70.5</td>
<td>86</td>
<td>82</td>
<td>40</td>
</tr>
<tr>
<td>Student</td>
<td>49</td>
<td>78</td>
<td>48</td>
<td>46</td>
<td>29</td>
<td>53</td>
</tr>
<tr>
<td>Student</td>
<td>95</td>
<td>88</td>
<td>78</td>
<td>54</td>
<td>39</td>
<td>20</td>
</tr>
<tr>
<td>Student</td>
<td>72</td>
<td>80</td>
<td>74</td>
<td>67</td>
<td>45</td>
<td>0</td>
</tr>
<tr>
<td>Student</td>
<td>70</td>
<td>94</td>
<td>75</td>
<td>66</td>
<td>63</td>
<td>0</td>
</tr>
<tr>
<td>Student</td>
<td>48</td>
<td>86</td>
<td>75.5</td>
<td>79</td>
<td>80</td>
<td>36</td>
</tr>
<tr>
<td>Student</td>
<td>63</td>
<td>82</td>
<td>76</td>
<td>79</td>
<td>68</td>
<td>62</td>
</tr>
<tr>
<td>Student</td>
<td>95</td>
<td>86</td>
<td>79</td>
<td>53</td>
<td>42</td>
<td>46</td>
</tr>
<tr>
<td>Student</td>
<td>74</td>
<td>60</td>
<td>49</td>
<td>75</td>
<td>64</td>
<td>47</td>
</tr>
</tbody>
</table>

**Fig. 1.** Comparative marks scored by students (left) and ChatGPT-generated scripts (right).

The second subsection examines the identification of ChatGPT-generated scripts by ALs. The remaining subsections consider the key plagiarism practices identified by the ALs in interviews.

### 4.1 Marking Performance

The student and ChatGPT-generated scripts for four modules were marked by ALs, with CS2 and CS3 being independently double marked. The final marks are shown in Table 5 and Figure 1.

In the case of the undergraduate modules (CS1, CS2, and CS3), every single ChatGPT-generated script achieved at least a passing grade (>40%). Every ChatGPT-generated script for CS4 was marked as a “fail” (postgraduate modules at the Open University have a higher pass threshold of 50%).
Within the passing grade, the Open University may choose to award a “distinction” grade for especially high-scoring student scripts. This is typically awarded for scripts scoring greater than 85%. Distinction grades would have been awarded to all of the CS1 ChatGPT-generated scripts. These results indicate that a student wishing to cheat by using ChatGPT to generate the entirety of their assignment could expect to pass these end assessments for CS1, CS2, and CS3.

As a postgraduate module, not only does the assessment of CS4 require deeper knowledge of topics than the other modules, but also it expects students to demonstrate greater proficiency of high-level learning skills such as synthesis and application of knowledge. A key aspect of the module’s assessment is the expectation that students will apply (and reference) the theoretical knowledge found in the module materials to an organization of their choosing. This combination of deep subject-specific knowledge, high-level learning skills, and intense personalization of solutions proved challenging for ChatGPT, which produced unsatisfactory superficial and generic answers to the assessment.

A key part of module assessment is the determination of final grades. Realizing that any form of marking involves subjectivity on the part of a marker, module teams at the Open University may choose to re-mark scripts just below grade boundaries to decide whether a higher grade is deserved. Therefore, script documents lying close to these boundaries receive greater scrutiny. However, since most ChatGPT-generated scripts received scores away from boundary grades, had they been submitted as genuine assignment documents, they would not be further scrutinized before being awarded a grade. Perhaps the most significant finding is not that ChatGPT behaves as an outstanding student across some of the undergraduate modules, but that it performs consistently as an “adequate undergraduate student”—able to pass assessment without drawing undue attention to itself.

### 4.1.1 Comparison of the Script Sample against Cohort Norms

Having established that most ChatGPT-generated scripts received passing marks, it is worth examining how representative the sample scripts—both ChatGPT generated and student—were to the wider cohort for those module presentations. Table 6 shows that for CS1, CS2, and CS3, the mean and standard deviations for marks align broadly with mean marks for the wider cohort, with mean ChatGPT-generated and sample student script scores lying within one standard deviation of the cohort mean. For CS1, the ChatGPT-generated scripts outperformed both the cohort and student sample, while ChatGPT underperformed relative to both for CS2 and CS3.

Once again, CS4 represents an outlier; not only did ChatGPT perform poorly in the assessment, but also the student sample is unrepresentative of the wider CS4 cohort. This does not affect the validity of the results for the ChatGPT-generated CS4 script documents, but it does prevent broader comparisons of the performance of the scripts against the cohort as a whole.

### Table 6. Performance of the ChatGPT-generated and Student Scripts in the Experiment against That of the Entire 21J/K Cohort for the Selected Modules

<table>
<thead>
<tr>
<th>Module</th>
<th>21J/K Cohort</th>
<th>ChatGPT Generated</th>
<th>Student</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cohort Size*</td>
<td>Mean Standard Deviation</td>
<td>Mean Standard Deviation</td>
</tr>
<tr>
<td>CS1</td>
<td>1,317</td>
<td>79.03 16.85</td>
<td>83.8 2.49</td>
</tr>
<tr>
<td>CS2</td>
<td>925</td>
<td>73.7 20.13</td>
<td>62.6 6.82</td>
</tr>
<tr>
<td>CS3</td>
<td>195</td>
<td>64.91 20.5</td>
<td>56.4 9.97</td>
</tr>
<tr>
<td>CS4</td>
<td>73</td>
<td>62.62 11.05</td>
<td>25.4 12.42</td>
</tr>
</tbody>
</table>

*Cohort size represents the number of students submitting their final assessment.
Table 7. Range of Total Marks Awarded to ChatGPT-generated Scripts and Students for Each Module

<table>
<thead>
<tr>
<th>Marker</th>
<th>ChatGPT-generated Scripts</th>
<th>Student Scripts</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>6.00</td>
<td>47.00</td>
</tr>
<tr>
<td>M2</td>
<td>13.08</td>
<td>28.46</td>
</tr>
<tr>
<td>M3</td>
<td>13.08</td>
<td>28.46</td>
</tr>
<tr>
<td>M4</td>
<td>4.00</td>
<td>40.00</td>
</tr>
<tr>
<td>M5</td>
<td>7.00</td>
<td>53.00</td>
</tr>
<tr>
<td>M6</td>
<td>28.00</td>
<td>88.00</td>
</tr>
</tbody>
</table>

Table 7 shows the range of marks awarded for ChatGPT-generated and student scripts. In each case, the ChatGPT-generated scripts received a lower range of marks than the genuine student scripts from the markers. Since each of a module’s ChatGPT-generated scripts was solely generated from identical prompts, there was limited scope for ChatGPT to generate radically different outputs. As previously stated, we did not edit the output of ChatGPT when creating the ChatGPT-generated scripts, so this low level of mark range may not be representative of actual cheating behavior where students may choose to supplement or alter the LLM’s outputs.

Again, CS4 is an exception. The extremely large spread in student scripts on the module is in good part due to two student scripts being awarded zero by the marker. In one of these cases, the student had erroneously submitted a script to a different piece of assessment; in the second, the researchers believe the script mark was erroneously transcribed as zero and should have been awarded 16%.

4.1.2 ChatGPT Performance on Individual Questions. We wanted to examine the marks awarded to individual questions to identify strengths and weaknesses in terms of the nature of topics being assessed. To do this, we categorized the questions by topic, examined the marks, and color-coded the question into six bands:

1. All scripts received a distinction mark for this question.
2. A majority of scripts received a distinction mark for this question.
3. A majority of scripts received passing marks for this question.
4. A majority of scripts received a fail mark for this question.
5. All scripts received a fail mark for this question.

Please see Appendix A.1 for a complete breakdown for all questions in CS1, CS2, and CS3. CS4 was not analyzed as the assessment is a single extended report, and therefore is not suitable for a question-level analysis.

Based on this analysis, we loosely identified some trends. For CS1, questions requiring a discussion of program development improvements and definition questions regarding the usage/vulnerabilities of SQL all received a distinction mark. For the questions focused on security and hashing, which involved stating definitions and applying techniques to simple scenarios, a majority of scripts received a distinction mark. The lowest-scoring questions—receiving passing marks—assessed programming and program development; an essay on social, legal, and ethical issues around digital literacy; and reflecting on module performance.

For CS2, there is mixed performance across the questions. The essay question on the use of robotics in space exploration, including sourcing examples with references, was answered well, with a majority of scripts receiving a distinction mark. The essay question on operating systems, building on a video on the history of Unix, received passing marks. Broadly, the three
short-answer opinion questions on operating systems, robotics, and networking also received passing marks. The personal development planning (PDP) questions were mostly handled poorly, with all the ePortfolio question responses failing, the majority of the self-reflection questions failing, and the future planning tending to receive passing marks. We did not use one of the optional long-form questions, as it was deemed unlikely that ChatGPT would be able to handle the practical networking activity using a network simulator without a huge amount of editorial work around the questions. This is consistent with our naïve approach of generating the ChatGPT scripts, as adapting the question would require too much effort from a naïve student.

For CS3, there is a significant consistent discrepancy between the two markers. That said, all 15 of the short-form questions requiring students to apply their understanding of a key HCI concept—from requirement gathering to design techniques and evaluation—broadly received passing or higher marks from one marker and majority distinction or higher from the other marker. For the long-form scenario question, the first part, which covers the development of interview questions, received passing or higher marks. The second part, requiring a heuristic evaluation of several interface screenshots, could not be passed through ChatGPT. The third and final part, requiring a redesign of an interface building on the heuristic evaluation, received passing or lower marks.

While it is obviously challenging for any given question to identify whether the question format or the topic covered was responsible or not for the ability of ChatGPT to generate compelling answers, this analysis demonstrates that, in most cases, across a range of question formats, topics, and study levels, ChatGPT is at least capable of producing adequate scripts.

### 4.2 ChatGPT-generated Script Identification

One of the most significant observations from the study was the distinction between markers’ ability to recognize suspect scripts and their decisions about which scripts to flag to the university. Although the number of scripts flagged as “suspected of plagiarism” by the markers was small, at interview it was clear that their identification of ChatGPT-generated scripts was more accurate than the flags alone might suggest.

Table 8 shows the plagiarism flags that the markers entered in the formal marking table during the marking exercise. An additional nine scripts were identified during the interview as appearing to contain “unusual or unexpected material” without meeting the bar for plagiarism flagging. This is discussed in depth in Section 4.3.

Of the three flags on student scripts, one relates to a script flagged by the original marker for potential breaches of the university’s student code of conduct; this is one of two such cases among the scripts sent to markers. Both of the other scripts flagged by markers referred to scripts where two of the three short-form questions (each worth 5/100 marks) either rely on re-writing an external source or make too much use of quoted material. Neither of these scripts would likely be referred for an academic conduct investigation, given that the two long questions (30 marks each) contained no academic conduct issues, and the scripts scored 18% and 16%, respectively, on Turnitin. Overall, the ALs were effective in detecting plagiarism in student scripts.

<table>
<thead>
<tr>
<th>Flag Type</th>
<th>ChatGPT-generated Scripts</th>
<th>Student Scripts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plagiarism flag during marking</td>
<td>7 (9 additional scripts flagged at interview)</td>
<td>3</td>
</tr>
<tr>
<td>Not flagged</td>
<td>23</td>
<td>57</td>
</tr>
</tbody>
</table>
The ChatGPT scripts flagged on the marking table were identified by the markers for the following reasons:

- M1 commented on one ChatGPT-generated script that “Parts of the Questions 6 and 7 appear to be missing” and on a different script that “[this script] and [the other flagged script] are similar for Q. 2(ii) for example.”
- M2 noted on one ChatGPT-generated script: “A mixture of some very good elements, and some less good. Inconsistent across questions. In a TMA, this would raise concerns for me.”
- M3 identified four distinct scripts from CS2, which shared a common, unusual approach to answering the same question, commenting: “Strange angle to answer - would swear it came from somewhere,” particularly in reference to the personal reflection element of the exam. These four scripts were identified as being unusual by M3, but the marker did not feel they met the threshold to raise a plagiarism flag.

From the in-script comments left by these markers, it is clear in each case that the questions raising concerns required students to provide either a personal viewpoint or some self-reflection. Either the answers provided are missing that sense of personal reflection (as with CS1) or a ChatGPT-generated script provides an answer written in the third person, talks about reflection in broad terms, and doesn’t relate back to an individual experience. The next section builds on this analysis by considering the data from the interviews conducted with markers, examining their practices regarding academic misconduct more broadly.

At the end of the interview, we informed the markers that five of their scripts had been authored by ChatGPT. None of them were surprised that some of the scripts had been generated by ChatGPT. The markers were asked to revisit their identification of ChatGPT-generated scripts. All but one of the markers re-identified scripts they had previously flagged or identified as containing unusual or unexpected material, with the same justification. The two CS2 markers had, between them, flagged all five ChatGPT-generated scripts for that module; at interview, M2 increased identification from one flagged during marking to correctly identifying all five. M6 hadn’t flagged any of the scripts for plagiarism in the marking feedback but immediately identified three of the ChatGPT-generated scripts (1, 10, and 11) and on reflection identified the two others (5 and 15). M6 also raised flags on some of the student scripts, likely due in part to the selection of student scripts not being reflective of the typical range (see Section 4.1.1). The only CS3 marker we interviewed—M4—responded by discussing suspicious symptoms rather than identifying specific scripts.

Overall, markers’ awareness of scripts containing unusual or unexpected material and their articulation of the characteristics that prompted suspicion were impressive, although their formal flagging of scripts was extremely selective.

4.3 Interview Analysis

We asked the interviewees about their practices regarding academic conduct: how they identified scripts of concern and how they decided whether to flag them. Four of the markers (M1, M2, M3, M4) highlighted a general expectation that plagiarism would be detected by either central members of the module team or the university’s anti-plagiarism software. M3 wrote: “you always say to yourself why should I flag it up? Because it goes through three or four pieces of software anyway, and they’ll flag it up.” This is a perfectly justifiable position, given a potential lack of understanding of central processes, limited time available for marking, and the range of different academic conduct issues that could be investigated: “there’s so much out there I could spend a lot of time looking” (M3).

Markers M3 and M4 both highlighted that they tended to have an extremely high threshold for academic conduct, seeing such situations as an opportunity for teaching and developing students.
M4 commented: “I haven’t flagged it with anybody else unless it’s blatant. I tend to actually put it to the students, you need to ... put this in as a reference. And this means not just in the references at the bottom, but in-text citations to say that this is where I got this information from, and I tend to flag it to the student in that way.” One of the reasons for this approach was articulated by M3: a wariness of mislabeling student work, due to both the impact on the student and potential consequences for the marker themselves.

Given the close relationship between students and markers, it is perhaps unsurprising—particularly in circumstances where marking time is limited and robust procedures are in place for other module team members to assess misconduct—that markers focus principally on teaching, i.e., on helping students improve their practices, rather than defaulting to disciplinary procedures, unless the misconduct is both blatant and severe. While this is an exemplary practice, it is unclear how well it will serve if easily accessible LLM tools result in increased cheating.

The identification of potential cases of academic misconduct is especially difficult in a distance setting, where “there’s a fine line between collaboration, peer learning, and collusion. And that’s an interesting challenge” (M2). Given that much of the correspondence and interaction between students, and between students and staff, occurs asynchronously, typically online and often on third-party unmonitored services such as private Facebook groups, it can be challenging to work out where to draw a boundary between acceptably collaborative learning and outright plagiarism. Current AL practice—as noted previously—tends toward providing study skills support rather than activating disciplinary procedures, although this did depend partially on the module level.

The ALs were very clear in distinguishing between aspects of student submissions that served as flags of misconduct and aspects that were the result of typical student behavior.

The most-mentioned flag was a change in style in answers, potentially indicative of different authors. This could be differences in the layout of the document itself or changes in the use of language, such as changes in tone or voice, the use of technical vocabulary, the proficiency of writing, or the specificity of the answers given: “obviously if you’re reading through something and there are significant language differences between answers to different questions” [M4]. This was particularly the case with changes in the use of technical language: “[where previous answers were] very general and didn’t know any technical detail, and suddenly you get an answer that’s full of technical detail, and something like that makes you very suspicious” [M3].

This sense of consistency as a key authenticity indicator led some tutors to gain a sense of the student voice by consecutively marking each complete script rather than marking each question in turn across all scripts. Similarly, as part of their usual marking practice, at least four of the markers would have looked back at students’ previously submitted work, as “you can spot that there’s something really going wrong, this student is totally, totally different” [M3]. However, as M2 noted, while consistency matters, sometimes it’s hard to identify issues as “they’ve just answered these questions on different days and didn’t proofread.”

M1 noted that many of the stylistic flags were more pronounced in material requiring students to integrate external material, noting that, particularly at Level 1, “if they’re asked to read a paper and they don’t actually put things sufficiently into their own words.…”

As the final flag, M6 noted that repetition acts as a significant flag: “it was the repetition that set me off”. Given the long-form essay style of this question in which a strong narrative element is expected, it is perhaps unsurprising that such a behavioral pattern, typified by ChatGPT’s often formulaic responses, acts as a clear flag to markers.

The markers also identified student behaviors that while not flags for misconduct were baseline behaviors requiring study support. In introductory modules, tutors are more forgiving as “I think their quality of academic writing tends to be flakier because they just don’t have the experience in it” (M2).
M4 identified the key points of assessment, namely: “One is, do you know what you’re talking about? Two is can you apply it? Three is can you communicate that information in an effective way?” This was deemed challenging to use as a flag, as students display various sub-optimal behaviors. These include:

- Not answering the question: “ohh, it’s about this. I’ll just tell you everything I know about this. Especially in an exam situation; you’re more likely to do that rather than stop and think” (M4).
- Not showing work: “if they’ve done a calculation and they have not bothered to put any working in, and then suddenly there’s one with lot of working” (M2).
- Not applying the scenario context to the answer given: “I did find a lot of them weren’t really utilising the scenario, the context they were given” (M4).
- Student performance being extremely variable: “the quality of students’ work is often quite variable” (M2).

Recognizing that these are familiar student behaviors is limiting, since marking feedback for the ChatGPT-generated scripts shows similar patterns of behavior. Hence, these behavioral cues cannot be used to distinguish between student scripts and ChatGPT-generated scripts without additional evidence.

The version of ChatGPT used in this study typically generates false references [53]; the research team examined every reference provided in the ChatGPT-generated scripts and found that all of the references that were included either were artificial or drew on material in the question used in the original prompt. We specifically asked markers about their behavior regarding checking references. Referencing has greater prominence in some modules (e.g., CS4 requires in-depth discussion of student-selected references; CS3 requires no references). Unsurprisingly, ALs pay greater or less attention to referencing based on the module they are marking. Both M4 and M2 noted that they typically scanned the formatting and venue of the reference for correctness but didn’t check them for validity as the time for marking is so tight. M3 was similar but noted that they would follow the references if they suspected plagiarism. Both M1 and M6 said that they would follow references, with the CS4 tutor (M6) noting that based on the module material, “if you can’t find a link to them, you can’t identify them, then it’s an automatic fail.” This helps account for the low scores received by many of the CS4 marked scripts and why it didn’t result in the scripts being flagged for plagiarism.

5 ACADEMIC MISCONDUCT SOFTWARE RESULTS

In addition to the marking exercise, we also examined how two key pieces of software coped with our script sample. We wanted to explore how accurately the scripts in our study (40 student scripts, 20 ChatGPT generated) could be identified by detection software. We recognize that we have no absolute guarantee that any of the student scripts were actually written by a person; however, these scripts were received by the university prior to the public release of ChatGPT. On the balance of probability—given the relative complexities of using previous-generation LLMs—we believe that all 40 student scripts were written by people but acknowledge that this may not be the case.

5.1 GPT-2 Detection

We ran all 60 scripts through the GPT-2 detection software. Figure 2 shows the results; the ChatGPT-generated scripts have been grouped on the right of the chart, with the student scripts on the left. We suspect a script of being of AI origin if it scored more than 0.2.

All of the ChatGPT-generated scripts score greater than 0.2, indicating they are likely to have been generated by an LLM. No student script scored above 0.5. Five student-generated CS1 scripts
scored >0.2, no student-generated CS2 scripts scored >0.2, four student-generated CS3 scripts scored >0.2, and two student-generated CS4 scripts scored >0.2. This represents a false-positive rate of between 0% and 50%.

5.2 TurnItIn AI Detection

Figure 3 shows that more recent detection tools—such as TurnItIn AI detection (trained on the GPT-3 and GPT-3.5 language models underpinning ChatGPT)—are far more successful at identifying ChatGPT-generated material. The mean percentage of text in student scripts considered of AI origin was 0.25% (SD of 1.17%, maximum of 8%). Comparatively, the mean for our ChatGPT-generated scripts was 74.43% (SD of 25.8%, minimum 28%). The detection rate was 100%, with a 0% false-positive rate.
6 DISCUSSION

From the time of Plato onwards, there has been a long history of resistance to new technologies in learning, including to the incorporation of calculators in the curriculum and the use of spellcheckers and word processors to improve the quality of submitted work. The debate on digital technology in education has a range, from those who think universal access to such technologies solves many problems to those who think misinformed use of digital technology robs education of essential human values [50]. Our work demonstrates that similar resistance to the use of generative AI tools in education is justifiable if assessment methods do not take account of this seismic change. Assessment material for varied standard topics from across the undergraduate computing curriculum all appear to be answerable by ChatGPT.

The discussion of our results is structured into three sections. In the first we consider our first RQ on the performance of ChatGPT on computing end-of-module assessments and how our findings contribute to an ongoing series of studies demonstrating that existing assessment practices within computing are vulnerable to academic misconduct from improper use of LLMs. We then consider our second and third RQs regarding the detection of LLM-generated material and what this means for assessment practices. Finally, we broaden our perspective to consider what our results mean for the future of assessment design within computing curricula.

6.1 ChatGPT Performance on CS End-of-module Assessment

Every single ChatGPT-generated script for the undergraduate modules (CS1, CS2, and CS3) received at least a passing grade (>40%), with all of the introductory module CS1 scripts receiving a distinction (>85%). Our question-level breakdown highlighted that the only question type that received consistently failing marks (<40%) were PDP and reflection-based questions.

Our results are broadly consistent with the few published studies examining other aspects of the curriculum including software testing [30], programming [19, 20], and security [33]. While there are significant differences between our work and these previous studies—particularly the use of double-anonymous markers outside of the research team and the question-level analysis—the consistent pattern is that ChatGPT and other LLMs are capable of producing content sufficient to pass undergraduate-level assessment in diverse curriculum areas. Some areas of curriculum remain unexplored—such as database design or networking—and there is a need for further work to understand whether there are topics where assessment remains resistant to LLMs. There is also a need to consider how we approach assessing undergraduate students.

The limited data we have for postgraduate material is that the version of ChatGPT used in this study could not produce material sufficient to pass the module. Our data is indicative that the higher-level cognitive skills we expect of postgraduate students [2] cannot be replicated with ChatGPT, but this is too small a dataset on which to base judgments, particularly given the unknown capacity of more advanced tools—such as later developments of ChatGPT based on GPT-4—to produce better answers to assessment material.

6.2 Detection of LLM-generated Material

Overall, our markers demonstrated impressive awareness of scripts containing unusual or unexpected material and were able to articulate the characteristics that prompted suspicion, although their formal flagging of scripts was extremely selective. This was indicative of an attitude viewing low-level potential misconduct as a teaching opportunity rather than a disciplinary offense, with a high bar set for disciplinary action that may result in potential serious consequences for students accused of breaching academic codes.

Perhaps surprisingly, the deterrent effect of detection and punishment remains unclear. Based on an analysis of collaboration, collusion, and plagiarism in computer science coursework, Fraser
argues that high rates of detection and prosecution deter cheats, and that students are more likely to cheat if they believe cheating is commonplace [21]. An empirical study by Bennett found that punishment was a deterrent to major forms of plagiarism but not necessarily to minor offenses [7]. However, Sheard et al. found that fear of consequences did not appear to have any influence on the level of cheating [46].

Software detection of LLM-generated content demonstrated differing efficacies. While both tools correctly identified all of the ChatGPT scripts, the GPT-2 detector produced a 0% to 50% false-positive rate among student scripts, with TurnItIn recording a 0% false-positive rate. This aligns to the findings of Orenstrakh et al., with eight tools providing very different levels of accuracy and false positives [38].

The most likely use-case at the Open University is for suitable detection tools working alongside feedback from ALs to select students for an oral examination that would resolve possible academic conduct issues. However, institutions would need to carefully balance the biases that could be generated by using the tools for detection. In other domains, early research is indicative that GPT detectors are biased against non-native English writers [31], significantly weakening the case for their use. Similarly, while the results from the TurnItIn AI detection software clearly distinguish student scripts from ChatGPT-generated scripts, its creators warn that its results are purely interpretive and should not be used as the basis for bringing disciplinary cases against students. It may also be necessary to have a policy of performing additional oral examinations on students who are not suspected of cheating so as not to unfairly stigmatize students who are initially flagged but following the oral examination are subsequently cleared of any wrongdoing.

Furthermore, detection software cannot take account of other concerns around the use of ChatGPT, including a student re-writing the material or a student passing the generated material through other software (such as Grammarly or other text adaption software) to rephrase the artificial text. As Rodriguez et al. note, the integration of ChatGPT-generated texts into a document containing some student-originated material is a common form of cheating that is much harder to detect through software [43]. Our baseline data highlights how successful the tools are for the most simplistic form of cheating.

In short, our findings can be summarized as “beware of snake oil”—technology alone cannot solve this problem. As the UK Quality Assurance Agency for Higher Education states:

“Detection tools - be cautious in your use of tools that claim to detect text generated by AI and advise staff of the institutional position. The output from these tools is unverified and there is evidence that some text generated by AI evades detection. In addition, students may not have given permission to upload their work to these tools or agreed how their data will be stored” [41].

Whatever improvements are made to detection systems, some of the concerns raised—particularly the lack of concrete proof—indicates that detection software is not a panacea for quality assurance issues raised by ChatGPT and LLMs.

6.3 Assessment Design

If LLMs can generate passable answers for unproctored undergraduate-level assessments in our discipline and detection software and markers cannot adequately distinguish LLM-generated material from student-authored material, we need to redesign our assessments.

In terms of insights into how to structure assessment, our analysis indicates that current LLMs performed relatively poorly with questions requiring a high level of reflective content or requiring greater cognitive skills (as with the postgraduate module CS4). One solution to the threat to academic integrity posed by LLMs is to expect more from our students—to worry less about the mechanistic aspects of our discipline and concern ourselves more with students being able to apply their
diverse skills in broad scenarios. This is not a novel suggestion—many institutes have such cross-over “authentic” assessments—but it would be a radical change to move all assessments to this model. Such a change would also require adaptations from accreditation organizations (such as the BCS [58]), which currently require definitive evidence that students have a certain set of competencies. This would be challenging to guarantee given the open-ended nature of such assessment.

Our data also showed that ChatGPT struggled to answer questions based around specific content not available outside of the Open University materials. This is unlikely to remain relevant with the release of portable “small LLMs” [47] that can be installed on personal computers and trained with specific datasets that could easily include Open University materials. There has been some success in using changelogs and behavioral detectors to monitor for plagiarism in programming [26], but it is unclear how successful this could be in other areas of the curriculum or whether it could be effective in detecting the use of LLM-generated content.

The proctored exam format has survived because it is efficient to administer and cheap to mark, and the controlled examination environment creates challenges for those choosing to cheat or plagiarize work. Unfortunately, many areas of computing are poorly suited to this exam format; for example, an understanding of programming is best examined through practical activities on a computer [15]. Furthermore, written final exams are known to cause anxiety among some students, while open-book exams can reduce anxiety and stress [42]. In such cases, highlighting institutional plagiarism policies and highlighting academic good practice has been associated with reduced plagiarism in CS courses [32, 34].

However, to ensure student authorship, at-home open-book online exams require online proctoring. Online proctoring has been strongly contested as a gross, and illegal, violation of privacy [9] as well as discriminatory, by assuming students have access to a reasonably private place where they can work on an exam without disturbance or violating the rights of others. As Swauger explains, there are concerns that proctoring technologies employing biometric techniques (e.g., facial recognition) disproportionately disadvantage minority students [49].

Additionally, trials have demonstrated that AI proctoring software is a poor detector of cheating behavior. Bergmans et al. [8] present a controlled experiment using the AI monitoring software Proctorio, which can flag “suspicious behavior” and alert a human invigilator to intervene. The experiment recruited 30 computer science students, 5 of whom were asked to behave nervously but complete the test honestly, and a further 6 of whom were asked to cheat. The system was ineffective at identifying cheating students, with nervous students being falsely flagged as cheating.

Replacing written assignments with oral examinations limit some of the risks LLMs pose. This has been demonstrated in CS, with mixed feedback from students (e.g., [17, 27, 42]). Motivated by reducing plagiarism in a system analysis and design course, Dick studied student interviews as a means of assessment [15] and found that performing interviews at two points during a semester-long project eliminated the student practice of copying assignments and disguising plagiarism by making minor changes. Further positive findings were that students received immediate feedback and had opportunities to practice and develop their communication skills. On the negative side, many students found the interview process stressful and expressed a strong dislike for this form of assessment.

Ohmann discovered that the students and instructors who participated in a final oral exam in a foundational CS course had positive reactions about their experience, with students demonstrating a deeper level of engagement with the material than previously noted with written assessment [37]. Regarding barriers to the implementation of oral exams, Ohmann cited difficulties with scaling the exam to a larger class, especially when single instructors do not have sufficient support from tutors or teaching assistants. Ohmann also highlighted a significant problem with students distributing exam questions to peers during the 5-day period required to complete every individual session.
6.4 Summary

There is a desperate need to assess how computing curricula should design new assessment strategies that encourage appropriate use of generative AI (to prepare students for the future) while ensuring academic standards are maintained. It is equally essential that best practice is rapidly disseminated. We do not proffer definitive answers but have outlined some of the options available based on reflections from our study data. The intention of our work was to assess the capabilities of ChatGPT—as an illustrative generative AI tool—to answer current assessment material. Our results help demonstrate the problem and the need to update assessments, assessment practices, and quality assurance practices. Our analysis demonstrates that across a range of curriculum areas, question formats, topics, and study levels, ChatGPT is at least capable of producing adequate scripts in undergraduate modules, and this is of concern for all educators. This issue requires ongoing reflection regarding academic integrity, curriculum design, and assessment strategies based on further developments of LLMs.

7 LIMITATIONS

7.1 Study Limitations

We recognize that this work has limitations worthy of further exploration. Many of these are specific to the design and execution of the study. First, four of the five markers we interviewed stated that they had an awareness of LLMs such as ChatGPT but had not used them. This lack of experience is likely to change over time as the tools become increasingly mainstream but likely had some form of impact on our results. We attempted to mitigate this impact by providing broad instructions to markers to report “anything unusual or unexpected found in the scripts.” While using prior experience of ChatGPT to filter applicants would have guaranteed that the markers had some understanding of the capabilities and output from LLMs, it would have removed the double-anonymous nature of the study, where markers were not aware of the purpose of the quality assurance exercise.

Second, while the ChatGPT-generated scripts often had a similar structure and tone of voice to one another, there were no exact matches. The study’s small scale means that insufficient ChatGPT-generated scripts to a single-question paper were generated to discover at what point limitations in ChatGPT would eventually produce essentially identical answers. When dealing with hundreds or thousands of ChatGPT-generated scripts on a limited topic, there may be a saturation point beyond which content is repeated—if content is repeated, quality assurance becomes an easier process.

Third, the ChatGPT-generated scripts produced in this experiment represent the laziest of cheats—those who take ChatGPT’s output and copy and paste it into their assignment answers without attempting to reword the answers or supplement the chatbot’s material with their own words. Any such edits or additions are almost certain to increase the difficulty of detecting cheating through both automated processes and the expertise of markers. This is an area ripe for further investigation.

Fourth, we only received a set of marks for a relatively small subset of a typical computing curriculum. We struggled to recruit markers for the four courses presented, and the advanced undergraduate and the postgraduate levels were each evaluated through a single module. We had prepared scripts for two intermediate undergraduate modules to which we failed to recruit any markers. We have no insights into how well suited ChatGPT and LLMs are for producing robust answers to other areas of the curriculum. This is a pragmatic deficiency and we urge others to make use of our methodology to continue contributing to the ongoing research and discussion in this area.
7.2 Limitations Inherent in Researching LLM Use

Beyond specific study design decisions, there are several limitations relating to difficulties in conducting educational research focused on LLMs.

One of the limitations of our work—which is reflective of the broader research space—is a result of the need to process student data with tools and technologies that are compliant with institutional policies toward ethical conduct and data protection as well as very strong UK legislation in these areas. For example, we were limited to using two LLM detection tools known to be dated since they were known to be compliant with all relevant institutional and national policies and legislation. A consistent compromise exists between ecological validity—in using real student data—and needing to understand the capabilities of tools and technologies that are inappropriate to use with student data. This means that even in the best-case scenario, educationalists are unlikely to be able to analyze data with the most recent and most powerful tools available.

Our work utilizes a single LLM, namely ChatGPT based on GPT-3.5. Since the work was undertaken, further capabilities have been embedded within later releases of ChatGPT, and other tools—such as Google Bard—have become more accessible. Therefore, our results represent a snapshot of what a certain LLM tool could achieve at a certain time. While we have no empirical data investigating these further capabilities, the likelihood is that the newer versions of LLMs have greater capabilities that increase the risks to academic integrity.

This has several implications in terms of interpreting the results and for other researchers in this area. The speed of change in LLM development compared to the slow publication schedule for much peer review work creates a tension. When interpreting results from peer-reviewed studies, readers must be aware that the capabilities of discussed technologies have likely improved or been entirely superseded and that the work can only represent a snapshot in time. Authors need to be clear in their presentation of the results that their findings represent a snapshot in a rapidly advancing field.

Replication of work is challenging, with some technologies rapidly improving and other technologies operating within closed boxes. This exacerbates the existing acknowledged crises in replication studies within computing education research [1, 24], which existing strategies—such as editorial policies and a healthy environment of replication studies—will not address [24].

The inertia in publications compromises the velocity of dissemination of new findings. Research teams are likely to be working on outdated or outmoded working hypotheses within fast-moving areas. For educationalists, this creates a necessity not only to remain aware of the academic literature but also to maintain an awareness of pre-prints and the “grey literature.” This may lead to a compromise of normal academic practice in order to expedite the dissemination of new findings through informally published materials, albeit with the risk that some referenced material may be substantially altered as part of a later, more formal publication process or withdrawn entirely.

8 CONCLUSIONS AND FURTHER WORK

This study has demonstrated that ChatGPT can achieve passing grades in several computer science undergraduate modules at different levels of study and with differing assessment models. Across a range of undergraduate modules, ChatGPT-generated scripts scored similarly to a random sample of genuine students. ChatGPT performed much less well in the single postgraduate module that was examined, with none of the ChatGPT-generated scripts achieving a passing mark. The study showed that it is not necessary for students to augment ChatGPT responses with their own work or edit its outputs to reach at least a passing grade; it is highly likely that students who supplement ChatGPT material would score even higher on their assessment.

Foundational-level assessment is extremely vulnerable to ChatGPT. Short-form questions based on simple definitions and applications of knowledge can be answered to a very high level without...
requiring extensive domain knowledge or skills in query generation. Likewise, questions containing detailed background information, those listing specific points to be addressed in a script, and those providing templated answers are amenable to high-scoring ChatGPT-generated scripts. While it is tempting to suggest these issues can be resolved by providing less guidance and requiring longer-form answers, such an approach seriously disadvantages novice students as well as those struggling with module content or language skills.

Our analysis demonstrates that across a range of question formats, topics, and study levels, ChatGPT is at least capable of producing adequate scripts for undergraduate modules—but it does provide better answers in some contexts. We need to better explore what assessment types for which topic areas are most resistant to generative AI to ensure academic quality assurance while continuing to support students’ learning.

In terms of detecting material generated by ChatGPT, our markers’ awareness of scripts containing unusual or unexpected material and their articulation of the characteristics that prompted suspicion were impressive, although their formal flagging of scripts was extremely selective. The software detection of LLM-generated content was successful at identifying all of the ChatGPT scripts, although the tools differed in their false-positive rate among student scripts. Our results indicate that detection techniques are not a panacea for the quality assurance issues raised by ChatGPT and LLMs. Regardless of what method of detection is used, some of the concerns raised—particularly in terms of false positives and the lack of evidence—appear insoluble.

The LLM challenge to academia strongly resembles that previously created by essay mills. However, the deployment of ChatGPT provides a new and distinct method for cheating, drives the financial cost to zero, greatly reduces the likelihood of being accused of plagiarizing content, and has reduced the time needed to produce scripts to a few minutes. Unlike essay mills, the speed and availability of ChatGPT allows its use in remote examination environments.

In the short term, a return to face-to-face examinations is a straightforward way of retaining credible assessment, with a longer-term strategy requiring a rethink of the role of assessment as well as how it is conducted, potentially including the greater use of viva-voce examinations, live presentations, and increased reliance on personal portfolios of work. As a discipline, we are uniquely well placed to assist in both informing institutional policy (as this work has at the Open University), and developing the necessary support material to teach students about the appropriate uses of generative AI. As a community, we also need to consider what competencies we expect from our computing students and share best practices as to how to make the best use of generative AI in both our teaching and assessment of those students.
A APPENDICES

A.1 Question Breakdown

Tables 9, 10, and 11 show the overall performance of ChatGPT-generated scripts in individual questions on CS1, CS2, and CS3. CS4 was not included in these diagrams since it is treated as a single long-form question.

Table 9. CS1 Performance

<table>
<thead>
<tr>
<th>Question</th>
<th>1</th>
<th>2a</th>
<th>2b</th>
<th>3a</th>
<th>3b</th>
<th>3c</th>
<th>4</th>
<th>5a</th>
<th>5b</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Marks</strong></td>
<td>7</td>
<td>19</td>
<td>4</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>13</td>
<td>7</td>
<td>8</td>
<td>20</td>
<td>5</td>
</tr>
<tr>
<td>Hashing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Program development</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social, legal and ethical</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Module reflection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 10. CS2 Performance

<table>
<thead>
<tr>
<th>Question</th>
<th>1a</th>
<th>1b</th>
<th>1c</th>
<th>2a</th>
<th>2b</th>
<th>2c</th>
<th>3a</th>
<th>3b</th>
<th>3c</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>M2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>M3</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Marks</strong></td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>10</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>Networking</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operating systems</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Robotics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Networking</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operating systems</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Robotics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Module reflection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDP*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDP*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 11. CS3 Performance

<table>
<thead>
<tr>
<th>Question</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16 a</th>
<th>16 b</th>
<th>16 c</th>
</tr>
</thead>
<tbody>
<tr>
<td>M4 Marks</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>M5 Marks</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

- Green: All distinctions
- Yellow: Majority distinctions
- Orange: Majority passed
- Light blue: Majority failed
- Red: All failed
- Gray: Not attempted

ACKNOWLEDGMENTS

We would like to thank Amel Bennaceur, Cecilia Domingo-Merino, Lucas Anastasiou, Nitu Bharati, Venetia Brown, Yaw Buadu, Neil Jones, Jo Buxton, Carole Thornett, and Marion Sacharin for sharing their insights. We would also like to thank all of our colleagues who helped support this work, particularly those ALs who were involved in the project.

REFERENCES


Bob or Bot: Exploring ChatGPT’s Answers to University Computer Science Assessment


Received 30 May 2023; revised 31 October 2023; accepted 9 November 2023