Students' Perceptions of Paper-Based vs. Computer-Based Testing in an Introductory Programming Course

João Paulo Barros

Polytechnic Institute of Beja, Beja, Portugal UNINOVA-CTS, Monte de Caparica, Portugal
joao.barros@ipbeja.pt

Keywords: Assessment, Tests, Exams, Lab Exam, Programming, CS1.

Abstract: This paper examines the preferences of students regarding computer-based versus paper-based assessment, in an introductory computer programming course. Two groups of students were surveyed about their prefer- ence between paper-based and computer-based tests and respective rationale. All students had already been assessed: one group using two paper-based tests and the other group using two computer-based tests. Both groups expressed an overwhelming preference for computer-based tests independently of their previous pro- gramming experience. We conclude that, from the students' point of view, computer-based tests should be the used over paper-based ones for introductory programming courses. This adds to existing literature about computer-based testing of programming skills.

1 INTRODUCTION

The teaching and assessment of programming skills is still an important and difficult topic, as demon- strated by the continuing large number of articles on the subject, e.g. (**?**; **?**; **?**; **?**; **?**; **?**). Furthermore, the use computer-based tests has also been the subject of some research, but typically in the context of learn- ing results, e.g. (**?**; **?**; **?**). Despite the preference students often demonstrate towards computer-based tests, anecdotal evidence and some published work indicates that paper-based tests are still widely used in introductory computer programming courses e.g. (**?**; **?**). This is probably due to tradition, fraud preven- tion, and the additional human and physical resources needed to properly apply computer-based tests e.g. (**?**). This paper presents the results of a study where two groups of students, in an introductory program- ming course, were assessed about their preferences regarding both types of tests and also the reasons why they prefer one to the other. Both groups had already completed two tests: one group completed computer- based tests, the other completed paper-based tests. The results provide supplemental help when deciding about or confronted with the need to choose one type of assessment over the other.

The paper has the following structure: Section **??** presents the course were students were assessed, the related work and the hypotheses that motivated the study; Section ?? presents the used methodology and characterises the participants; Section ?? discusses the results and Section ?? concludes.

2 BACKGROUND

This section presents the course, the structure of the tests, the related work, and the research questions that we set out to answer.

2.1 Course Content and Structure

The course is the first programming course and is part of two computer science degrees in a small higher ed- ucation school. The course uses an objects-early ap- proach, the JavaTM programming language (?), and the BlueJ IDE (?; ?). First, students learn numeric types, arithmetic expressions, variables, constants, and the use of mathematical functions, by analogy with a scientific calculator. After they apply condi- tionals, loops, and vectors to make more complex cal- culations. Finally, they use graphical objects and re- cursion. The grading is based on individual tests and each test can improve the grade of the previous one, i.e., a better second grade will replace the first one. The same is done for each subsequent test. e.g. (?; ?).

2.2 Tests

The paper-based and the computer-based tests had an identical structure and content: students had to write small functions to compute numerical values, or to write number or text patterns using loops. For the paper-based tests the grading criteria was extremely tolerant regarding syntax errors and the students only had to write the core functions (no need to write the main method or imports). Even simple output er-rors were given a small penalty. Regarding computer-based tests, the students had to submit code without compilation errors. Code with compilation errors got a zero mark, just like non-delivered code. The correct output was the main criteria. Wrong output implied a strong penalty, even with a near correct logic.

2.3 Related Work

The importance of computer-based assessment in in-troductory programming is recognised for quite some time. Daly and Waldron concluded that the computer-based tests (lab exams) are more accurate assessors of programming ability than written exams or program-ming assignments (?). Yet, it is also know that its effective application is more demanding than paper-based assessment. This is attested by Bennedsen and Caspersen (?) where students are assessed by a computer-based test but in small groups, with two teachers in the room and only for 30 minutes. (?) concluded that computer-based tests effective at in-creasing student motivation even over group assign-ments. (?) have found that for a specific programming problem, when students were allowed to use the com-puter to continue a paper-based test they were able to correct remaining errors in the respective programs.(?) found out that students who took a computer-based exam to write a recursive solution to a binary tree operation were more successful than those who took the paper-based exams (58% vs. 17% correct solutions). Rajala et al. present the adaptation of au-tomatically assessed electronic exams and note that computer-based exams have potential benefits for stu-dents, including, for example, the possibility to com-pile, test and debug the program code. They recom-mend computer-based exams for other educators as well (?).

2.4 Research Questions

The research questions were motivated by anecdotal evidence as students seemed to almost always, with very few exceptions, prefer computer-based tests.
Also, due to insufficient human and physical re-sources, we were forced to apply paper-based tests to one group of students, while the remaining ones, in the same course, completed computer-based tests. Hence, we decided to ask both groups of students about what kind of tests they prefer. Then, with the intention of exposing students to the perceived advan-tages of each type of test, they were asked to select from a list the advantages of each approach. Each student could also point out additional advantages for one or both approaches. Hence, the research ques-tions were the following: RQ1 Do students prefer computer-based tests over paper-based tests?
RQ2 What are the perceived advantages students find in each type of tests?
RQ3 Students' opinion changes after being con- The third research question (RQ3) was assessed by asking the first one (RQ1) before and after students were asked to point out the perceived advantages of each type of test.

3 METHOD OF STUDY

The method of study was an anonymous question-naire. All students were invited to complete it. The invitation was by a post (delivered by email) on the course forum. There were two additional reminders to answer the questionnaire with a three days dead-line. The students were divided in two groups (A and B) and the same questionnaire was applied to both:
Group A The students who had completed
computer-based tests;
Group B The students who had completed paper-
based tests.
First, students were asked about their previous programming experience to allow checking eventual differences in preferences between them. Then, the following slider scale was used. An even number of options was used to force the respondents to choose between paper-based and computer-based testing, but in a non-binary way:
Question: In what measure do you prefer tests to be made in paper or computer?
Answer:

This slider scale provided the answer to RQ1. In the following question, students were asked to select items that contributed to the rationale for their prefer-ence, thus providing data for RQ2. From now on we name that question the ”rationale question”. After be-ing asked to make this selection (answering the ratio-nale question), students were asked the same question with the same slider scale. This provided the data for RQ3.

4 RESULTS AND DISCUSSION

The populations for Group A and B had a size of 92 and 21, respectively. For each group, the total num-ber of responses was 35 (38%) and 16 (76%), respec-tively. We used those as the two samples: Group A and Group B.

Figure ?? shows the frequencies for the prefer-ences from 1 (strong preference for paper-based tests) to 10 (strong preference for computer-based tests) be-fore and after the rationale question. It is very clear that both groups prefer computer-based tests. Inter-estingly, the group of students who in fact completed computer-based tests are even more in favour of that type of tests.

After the rationale question, it is possible to ob-serve a slight decrease in the highest preferences for computer-based tests. Table ??presents the mode and median for all the students in Group A and Group B, before and after the rationale question and it makes more evident that only the students completing paper-based tests become slightly less critical of those tests after answering the rationale question. Possibly, this is due to increased awareness about the perceived relative disadvantages of computer-based tests that resulted from the pondering over the advantages of paper-based tests in the rationale question.

Table ?? and ?? additionally show that this change, although very weak, is more pronounced in students with previous programming experience even for the ones already completing computer-based tests.

Figure ?? shows the percentage of students in each group that selected each one of the items in the ratio-nale question. The prefixes ”C” and ”P” identify al-leged advantages of computer and paper based tests, respectively. Students could also add other reasons, but only two students, both from Group B (doing paper-based tests) used that possibility: one said that ”with paper everything stays in the head”; this stu-dent was the strongest supporter of paper-based tests (having answered 1 both times) and had no previous programming experience; another students, this time a strong supporter of computer-based tests, added that ”in the computer I can add variables I had forgotten to add before”.

It is very clear that the advantages of computer-based tests are much more frequently pointed out. This is especially relevant as the question was writ-ten as ”Check what are, in your opinion, the advan-tages of paper-based tests and computer-based tests”. Hence, the students are much less willing to recognise the advantages of paper-based tests. Apparently, the preference for computer-based tests goes to the point of demotivating students to select the advantages of paper-based tests. In fact, even obvious advantages of paper-based tests like ”P - in the paper there is no risk of a computer malfunction” were chosen by only 19%of Group A and 31% of Group B.

Interestingly, the preference for ”copying code that it is possible to bring” to a computer-based test (in the context of an open book test) is arguably a dis-advantage of computer-based tests, as students, espe-cially weaker ones, tend to just copy paste some code and then try to solve the problem by trial and error. In simple problems they can even succeed without really understanding why or how the program really works. Finally, it is important to note that the significant difference in the sample sizes (group A and group B) and response rates are important limitations of this study. Besides larger and more similar group sizes, a more detailed characterisation of student background would be desirable. Yet, this may imply a non-anonymous questionnaire.

5 CONCLUSION

The study allowed us to conclude that students in our sample have an overwhelming preference for computer-based tests, to the point that they tend to re-sist recognising the advantages of paper-based tests. Students also maintain to a great extent their prefer-ence even after going through a list of advantages of one type of testing over the other. We believe this strong preference for computer-based tests has a sig-nificant effect in students’ motivation. In that sense, our study reinforces previous ones that pointed out the learning advantages of computer-based tests e.g. (?; ?).

As future work, we intend to ask third year stu-dents and also older students already in the work-place about their preferences regarding paper-based vs. computer-based tests. Also, it would be interest-ing to search for eventual correlations between perfor-mance in the tests and test style preferences. Finally, an interesting alternative approach would be to have all students experienced both test styles, in different order, before asking the preference.

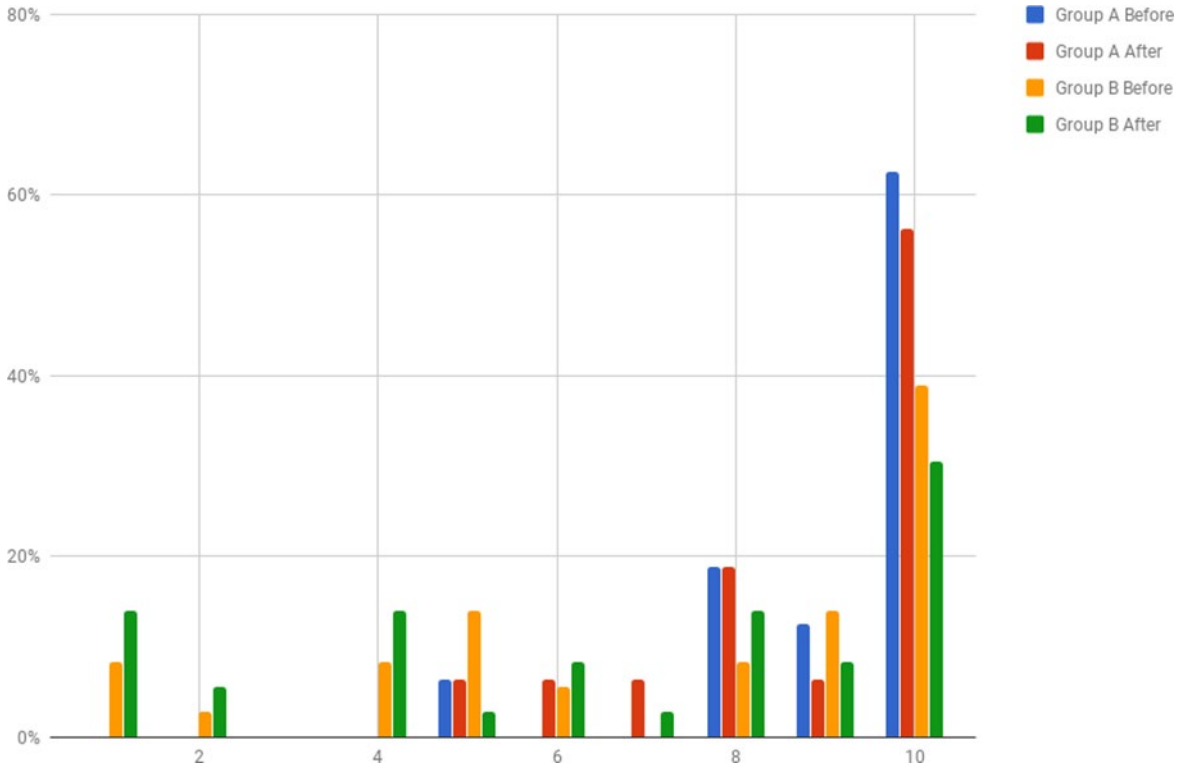

Figure 1: Percentage of chosen values in the scale 1 (strong preference for paper-based tests) to 10 (strong preference for computer-based tests).

Table 1: Expressed preferences for paper versus computer based tests, before and after choosing relative advantages.

Group A (completed computer tests)				Group B (completed paper tests)			
Mode		Median		Mode		Median	
Before	After	Before	After	Before	After	Before	After
10	10	10	10	10	10	9	8

Table 2: Expressed preferences for paper versus computer based tests, before and after choosing relative advantages by students with previous programming experience.

Group A (completed computer tests)				Group B (completed paper tests)			
Mode		Median		Mode		Median	
Before	After	Before	After	Before	After	Before	After
10	10	10	10	10	10	9	8.5

Table 3: Expressed preferences for paper versus computer based tests, before and after choosing relative advantages by students without previous programming experience.

Group A (completed computer tests)				Group B (completed paper tests)			
Mode		Median		Mode		Median	
Before	After	Before	After	Before	After	Before	After
10	8	9.5	8	10	10	8	7

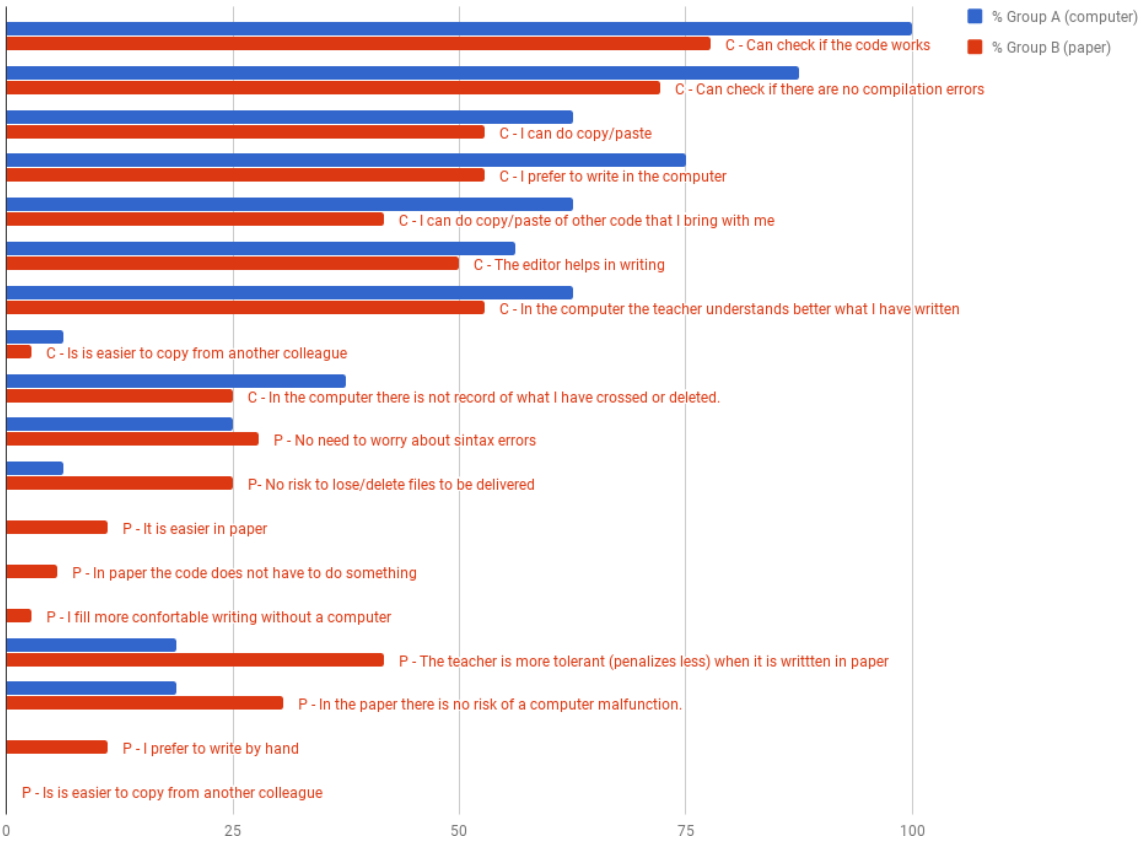

Figure 2: Percentage chosen for each item in the rationale question.

REFERENCES

Bain, G. and Barnes, I. (2014). Why is programming so hard to learn? In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, pages 356–356, New York, NY, USA. ACM.

Barros, J. (2010). Assessment and grading for CS1: To- wards a complete toolbox of criteria and techniques. In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, Koli Calling '10, pages 106–111, New York, NY, USA. ACM.

Barros, J. a. P., Estevens, L., Dias, R., Pais, R., and Soeiro, E. (2003). Using lab exams to ensure programming practice in an introductory programming course. In *Proceedings of the 8th Annual Conference on Inno- vation and Technology in Computer Science Educa- tion*, ITiCSE '03, pages 16–20, New York, NY, USA. ACM.

Bennedsen, J. and Caspersen, M. E. (2006). Assessing pro-cess and product – a practical lab exam for an introductory programming course. In *Proceedings. Fron- tiers in Education. 36th Annual Conference*, pages 16–21.

BlueJ (2016). BlueJ homepage. `http://www.{BlueJ}.org`. Accessed on 2017/02/09.

Chetty, J. and van der Westhuizen, D. (2015). Towards a pedagogical design for teaching novice programmers: Design-based research as an empirical determinant for success. In *Proceedings of the 15th Koli Calling Conference on Computing Education Research*, Koli Call- ing '15, pages 5–12, New York, NY, USA. ACM.

Daly, C. and Waldron, J. (2004). Assessing the assessment of programming ability. In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '04, pages 210–213, New York, NY, USA. ACM.

Grissom, S., Murphy, L., McCauley, R., and Fitzgerald, S. (2016). Paper vs. computer-based exams: A study of errors in recursive binary tree algorithms. In *Proceed- ings of the 47th ACM Technical Symposium on Computing Science Education*, SIGCSE '16, pages 6–11, New York, NY, USA. ACM.

Gmez-Albarrn, M. (2005). The teaching and learning of programming: A survey of supporting software tools.

The Computer Journal, 48(2):130–144.

Java (2017). JavaTM Programming Language. https://docs.oracle.com/javase/8/docs/technotes/guides/language/index.html. Ac- cessed on 2017/12/06.

Ko¨lling, M., Quig, B., Patterson, A., and Rosenberg, J. (2003). The BlueJ system and its pedagogy. Computer Science Education, 13(4):249–268.

Lappalainen, V., Lakanen, A.-J., and Ho¨gmander, H. (2016). Paper-based vs computer-based exams in CS1. In Proceedings of the 16th Koli Calling In- ternational Conference on Computing Education Re- search, Koli Calling '16, pages 172–173, New York, NY, USA. ACM.

Pears, A., Seidman, S., Malmi, L., Mannila, L., Adams, E., Bennedsen, J., Devlin, M., and Paterson, J. (2007). A survey of literature on the teaching of introductory programming. In Working Group Reports on ITiCSE on Innovation and Technology in Computer Science Education, ITiCSE-WGR '07, pages 204–223, New York, NY, USA. ACM.

Rajala, T., Kaila, E., Linde´n, R., Kurvinen, E., Lokkila, E., Laakso, M.-J., and Salakoski, T. (2016). Automatically assessed electronic exams in program- ming courses. In Proceedings of the Australasian Computer Science Week Multiconference, ACSW '16, pages 11:1–11:8, New York, NY, USA. ACM.

Sheard, J., Simon, Carbone, A., Chinn, D., Clear, T., Corney, M., D'Souza, D., Fenwick, J., Harland, J., Laakso, M.-J., and Teague, D. (2013). How difficult are exams?: A framework for assessing the complex- ity of introductory programming exams. In Proceed- ings of the Fifteenth Australasian Computing Educa- tion Conference - Volume 136, ACE '13, pages 145– 154, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

Silva-Maceda, G., David Arjona-Villicana, P., and Edgar Castillo-Barrera, F. (2016). More time or better tools? a large-scale retrospective comparison of pedagogical approaches to teach programming. IEEE Trans. on Educ., 59(4):274–281.

Simon, Chinn, D., de Raadt, M., Philpott, A., Sheard, J., Laakso, M.-J., D'Souza, D., Skene, J., Carbone, A., Clear, T., Lister, R., and Warburton, G. (2012). In- troductory programming: Examining the exams. In Proceedings of the Fourteenth Australasian Comput- ing Education Conference - Volume 123, ACE '12, pages 61–70, Darlinghurst, Australia, Australia. Aus- tralian Computer Society, Inc.

Vihavainen, A., Airaksinen, J., and Watson, C. (2014). A systematic review of approaches for teaching introductory programming and their influence on success. In Proceedings of the Tenth Annual Conference on International Computing Education Research, ICER '14, pages 19–26, New York, NY, USA. ACM.