Usability of Error Messages for Introductory Computer Science Students

Paul A. Schliep
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
schli202@morris.umn.edu

ABSTRACT

Error messages are an important tool for programmers to help find and fix errors in their code. When an error message is unhelpful, it can be difficult to understand how to find and fix the mistakes. Error messages are especially critical for introductory programmers in understanding problems with their code, but not all error messages are beneficial for helping novice programmers. This paper discusses how well error messages can help introductory-level computer science students resolve mistakes in their programs and what aspects make an error message more user-friendly for introductory computer science students. After that, we discuss the analyses of syntax, compiler, and exception errors and their results. I will also be discussing several methodologies and programs developed to help improve the experience a novice programmer has when attempting to understand causes for errors and their results.

Needs more work

Keywords

Novice programmers, usability, error messages, user-studies, compiler errors, exception errors, syntax errors, functional-programming

1. INTRODUCTION

One of the most important foundations of computer programming is the communication between the system and the user, specifically in the error messages produced by the system. These error messages are especially important for introductory-level computer science students to help them resolve issues in their program because they are the primary source for understanding what's wrong and "[they] lack the experience to decipher complicated or poorly-constructed feedback" [4]. The first rule of good message design is to be sure that the error doesn't add confusion [3], but when a novice programmer receives an error message that they can not understand, it becomes difficult to fix their program and often leads to frustration because the error message was either too complicated to understand or lead them down the wrong path [5] and can sometimes introduce new errors [1].

Several studies have been conducted on modern programming languages and IDEs error messages to study the effectiveness in helping novice programmers debug their program and help learn the concepts and programming languages. The results have shown that students struggle with compiler and syntax error messages [1] [7] (which will be defined in Section 2) and the general vocabulary of the error messages along with IDE-specific features such as source highlighting can be bothersome for introductory computer science students [5].

Several tools and heuristics are being developed to help address issues in error message usability and its development. The goals of these methodologies are to help introductory programmers learn the language and concepts easier. The goal of this paper is to discuss the analyses and their results of error message design and its usability for introductory students in a classroom setting, and how these developed methodologies help improve the user experience with error messages and their results.

This paper is divided into five sections. In Section 2 we discuss usability studies, define compiler, syntax, and exception error messages, and discuss imperative and functional programming. In Section 3 we will focus on analyses of the usability of exception messages and compiler messages for introductory students and how those analyses were performed. In Section 4 we discuss the results of those analyses. In Section 5 we discuss three methodologies developed to help improve the error message usability: Traver's heuristics for compiler message design, Marceau, Fisler, and Krishnamurthi's recommendations for error message design, and Denny, Luxton-Reilly, and Carpenter's syntax error enhancement tool [7] [5] [1].

Needs more work

2. BACKGROUND

This is where I will describe the concepts such as functional and imperative languages, usability studies, and error message types needed to explain the research done on error message usability. I will use multiple sources from primary and background to describe these concepts.

TODO

2.1 Runtime and Compilation errors

UMM CSci Senior Seminar Conference, December 2013 Morris, MN.

2.2 Imperative and Functional programming

3. ANALYSES

TODO

3.1 Analysis of exception messages

This will be a combination of two main sources used for this part: Mind Your Language: On Novices' Interactions with Error Messages and Measuring the Effectiveness of Error Messages Designed for Novice Programmers

3.2 Analysis of compiler messages

This will be a combination of two main sources used for this part: On Compiler Error Messages: What They Say and some of Enhancing Syntax Error Messages Appears Ineffectual

4. RESULTS OF CASE STUDIES

This will discuss the studies done on error message usability for introductory students and their results and what those results mean

TODO

5. METHODOLOGIES

Introduce methodologies meant to be used to improve error message libraries

TODO

5.1 Principles of compiler error design

This will use my source, On Compiler Error Messages: What They Say and What They Mean

5.2 Recommendations for error messages

This will use my source, Mind Your Language: On Novices' Interactions

5.3 Syntax error message enhancement and results

This will use my source, Enhancing Syntax Error Messages Appears Ineffectual

6. CONCLUSION

Discussion of the direction usability of error messages will be taken and how the methodologies will be applied for future work.

TODO

7. ACKNOWLEDGMENTS

TODO

8. REFERENCES

[1] Denny, P., Luxton-Reilly, A., and Carpenter, D. Enhancing syntax error messages appears ineffectual. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education* (New York, NY, USA, 2014), ITiCSE '14, ACM, pp. 273–278. *Discusses syntax errors and helps coincide with compiler source.*

[2] Hartmann, B., MacDougall, D., Brandt, J., and Klemmer, S. R. What would other programmers do: Suggesting solutions to error messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 1019–1028. *Presents a framework for adding hints for error messages, can probably mention hints in methodologies.*

[3] Isa, B. S., Boyle, J. M., Neal, A. S., and Simons, R. M. A methodology for objectively evaluating error messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1983), CHI '83, ACM, pp. 68–71. *can use this in methodologies and probably as a background in several other areas.*

[4] Marceau, G., Fisler, K., and Krishnamurthi, S. Measuring the effectiveness of error messages designed for novice programmers. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education* (New York, NY, USA, 2011), SIGCSE '11, ACM, pp. 499–504. *Helps with analyses and results of usability of error messages and presents a nice rubric for improving error messages.*

[5] Marceau, G., Fisler, K., and Krishnamurthi, S. Mind your language: On novices' interactions with error messages. In *Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software* (New York, NY, USA, 2011), Onward! 2011, ACM, pp. 3–18. *This paper along with the next help with a lot of the analyses and results of error message research.*

[6] Murphy, C., Kim, E., Kaiser, G., and Cannon, A. Backstop: A tool for debugging runtime errors. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education* (New York, NY, USA, 2008), SIGCSE '08, ACM, pp. 173–177. *Can possibly mention this in the conclusions as a background source.*

[7] Traver, V. J. On compiler error messages: What they say and what they mean. In *Advances in Human-Computer Interaction* (2010).

[7] [5] [4] [1] [2] [6] [3]