

# Homework 02 - Literature Search

Andrew Carter and Beryl Egerter

February 5, 2013

This paper is an overview of how concepts are relevant in program comprehension. It talks about both location of code concepts and how people learn from the code. It relates to our proposed study because we are interested in the process of program comprehension. [6]

This is a summary of six different program comprehension models and areas where they might not be as useful. This is useful information for our study as an explanation of theories of program comprehension that exist. [8]

This article talks about control structure diagrams, a specific method that can be used to aid program comprehension. This gives us a look at a solution to code comprehension. [3]

This article analyzes how programmers maintain unfamiliar code in an industrial setting. This gives a view of how program comprehension works in an industrial setting as opposed to an academic setting that we are doing our study in. [7]

This article proposes a course on code comprehension methods. Models used or concepts that are planned to be emphasized will likely be informative to our study. [1]

This article gives a theory of the code comprehension process, which may or may not be different than other theories mentioned in other papers we will read. [2]

This paper summarizes some terminology of reverse engineering or "program comprehension" as well as giving reasons why it is a difficult field. The paper may be useful orienting us to read other research in this field. [5]

This study looks at the effect of indentation in program comprehension. This will be good to keep in mind for our study as it brings up a number of small changes that could affect task completion over different problems. [4]

## References

- [1] III Austin, M.A. and M.H. Samadzadeh, *Software comprehension/maintenance: an introductory course*, Systems Engineering, 2005. ICSEng 2005. 18th International Conference on, aug. 2005, pp. 414 – 419.

- [2] Ruven Brooks, *Towards a theory of the comprehension of computer programs*, International Journal of Man-Machine Studies **18** (1983), no. 6, 543 – 554.
- [3] D. Hendrix, II Cross, J.H., and S. Maghsoodloo, *The effectiveness of control structure diagrams in source code comprehension activities*, Software Engineering, IEEE Transactions on **28** (2002), no. 5, 463 –477.
- [4] Richard J. Miara, Joyce A. Musselman, Juan A. Navarro, and Ben Shneiderman, *Program indentation and comprehensibility*, Commun. ACM **26** (1983), no. 11, 861–867.
- [5] Michael L. Nelson, *A survey of reverse engineering and program comprehension*, CoRR **abs/cs/0503068** (2005).
- [6] V. Rajlich and N. Wilde, *The role of concepts in program comprehension*, Program Comprehension, 2002. Proceedings. 10th International Workshop on, 2002, pp. 271 – 278.
- [7] A. von Mayrhauser and A.M. Vans, *Industrial experience with an integrated code comprehension model*, Software Engineering Journal **10** (1995), no. 5, 171 –182.
- [8] ———, *Program comprehension during software maintenance and evolution*, Computer **28** (1995), no. 8, 44 –55.