

Good Quality of Information Streams in Modula/Oberon Programs

Ivan S. Zapreev, Institute of Informatics Systems, Novosibirsk, Russia

There are different criteria of a program quality such as safety, stability, understandability, testability, good organization of information streams and flow of control [1], etc.

The aim of this work is to create algorithms that check quality of information streams in Oberon/ Modula programs depending on criteria of information streams regularity and confirmation [2] and a stream analysis. The stream analysis is an inter-procedural context sensitive analysis with approximation of mandatory and eventual informational relations [3] that implements an abstract interpretation of a program in the sense of Cousot [4]. The criterion of regularity is checked depending on information graphs of a program's linear parts. The criterion of confirmation is checked depending of the stream analysis results. This work also studies an approach to stream analysis results and criteria violations visualization.

Implementation of an analyzer that checks quality of information streams in Oberon/Modula programs is the part of the work. This analyzer has been built on the base of the WASP system (Oberon-2 / Modula-2 Static Analyzer) in the XDS environment (Linux/Windows 95/NT). A Graphic user interface has been developed to visualize stream analysis results and criteria violations. An XML based format has been developed to store analysis results.

References:

- [1] Igor V. Pottosin, A "Good Program": An Attempt at Exact Definition of the Term// Programming and Computer Software, v.23, № 2 1997, p. 59-69.
- [2] Igor V. Pottosin, Good quality of Programs and Information Streams// Open Systems, № 6 1998, p. 41-45.
- [3] Vladimir I. Shelechov, A program structure in the language-oriented stream analysis// Programming, № 3 1996, p. 47-59.
- [4] Cousot P. and Cousot R. Abstract Interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoint// Rec. of the 18th ACM Symposium on Principles of Programming Languages ACM Press, 1977, p.55-56.