

Translating Structured Data

Topic 21: Introduction to Compiler Design - Translating Structured Data

Petr Navrátil (xnavra53), Martin Lach (xlachm03)

31. 10. 2018

A compiler can encounter a lot of various variable types during compilation. Most programming languages provide floating-point numbers and structured values like arrays, records (structs), unions, lists or tree-structures [1]. This presentation will look into handling these types. One of the most important questions during the compilation is which address should be assigned to each variable. The compiler distinguishes between static and dynamic variables for which the address computation differs.

At first, the translation will be shown on the basic data types (integers, booleans, etc.).

Then the presentation will focus on structured data types. Starting with arrays – address computing, index checking, multidimensional arrays.

Custom user types (records/structs and unions) will be the last topic of the presentation.

References

- [1] Mogensen, T. Æ.: *Intermediate-Code Generation*. Cham: Springer International Publishing, 2017, ISBN 978-3-319-66966-3, s. 127–156, doi: 10.1007/978-3-319-66966-3_6.
URL https://doi.org/10.1007/978-3-319-66966-3_6