The effects of using exception handling on software complexity

Gergely Nagy, Zoltán Porkoláb

Eötvös Loránd University

`njeasus@ceasar.elte.hu, gsd@elte.hu`

Exception handling is the definitive way to handle errors of any kind and exceptional circumstances in modern software. There has been a long way before software methodlogy arrived to creating and using the notion of *exceptions*. We automatically assume that using exception handling makes our software more readable, more maintainable and easier to understand – i.e. less complex than when we use any other error management (let it be using return values, ERRNO or any other kind). Is this really the case?

Measuring *software complexity* can be done using software metrics. There are several trivial, well-known candidates – lines of code, cyclomatic complexity or McCabe-metrics and A-V – for this purpose.

In this paper, we extend the definitions of two metrics to the case of exceptions (lines of code is trivial in this matter) and analyze how these extensions affect what our metrics state about software products. We also examine real-world software to try to prove that our definitions have no negative effect on the complexity measured by these metrics.