Debugging

Søren Hansen <sha@ase.au.dk>
V1.0

Introduction

The *art of debugging* is actually a discipline which does not get the attention that it ought - why? In real life software project upward 50% of your time may be spend on debugging... So yeah having some sort of approach and understanding of the tools at hand matters - *alot!*.

Content and reflection

Themes

- Generally[2][3][1]
  Covers
  - How to go about debugging
  - Essential elements when debugging
  - Different error categories
- GDB[1][4, chapters 2-2.1.1, 4 (pp. 25), 4.10, 5-5.2, 20.1-20.3.4][5]
  Covers
  - How is gdb used
  - How do we do cross debugging
  - How is ddd used
  - What is a core dump and how is it used
- Tools for dynamic code analysis (*Valgrind's homepage - JFGI*)
  - valgrind - memory checker
  - helgrind - thread data and deadlocks checker
  - callgrind - profiler

Questions

- What are the 6 *essentials*
  - Consider each and discuss its merits
  - Which do you currently employ - if any :-)
- Error types / categories
  - What are they
  - Do you know how they present themselves in code - e.g. do you know how to find each and everyone (why is, in fact, extremely important that you know this???)
- What is a debugger is and how is used - at least from a fundamental point of view.

Material

Slides

[1] S. Hansen, *Debugging*, Slides - see course repos.

Local repository

[2] T. E. Boult, *Debugging techniques*, URL does not work(http://vast.uccs.edu/tboult/CS330/NOTES/debugging.ppt), but local file exists with the name: Debugging_presentation.pdf.

Online

[3] T. Parr. (2004). Learn the essentials of debugging, [Online]. Available: `http://parrt.cs.usfca.edu/doc/debugging.html`.

[4] R. e. a. Stallman. (). Debugging with gdb. Link to the most recent manual, [Online]. Available: `http://sourceware.org/gdb/current/onlinedocs/gdb.pdf.gz`.

[5] R. H. Pesch. (). Gdb quick reference. Link to the most recent manual, [Online]. Available: `http://sourceware.org/gdb/current/onlinedocs/refcard.pdf.gz`.