Dynamic Analysis

CSCI 3308 – Lab11

Program Analysis

Program analysis is a technique that reasons about the run-time behavior of the program

–Static program analysis – reasoning is done statically, before program execution

–Dynamic program analysis – reasoning is done dynamically, during program execution

GDB - GNU Debugger

GDB is the standard debugger for the GNU operating system. However, its use is not strictly limited to the GNU operating system; it is a portable debugger that runs on many Unix-like systems and works for many programming languages, including Ada, C, C++, Objective-C, Free Pascal, Fortran, Java and partially others.

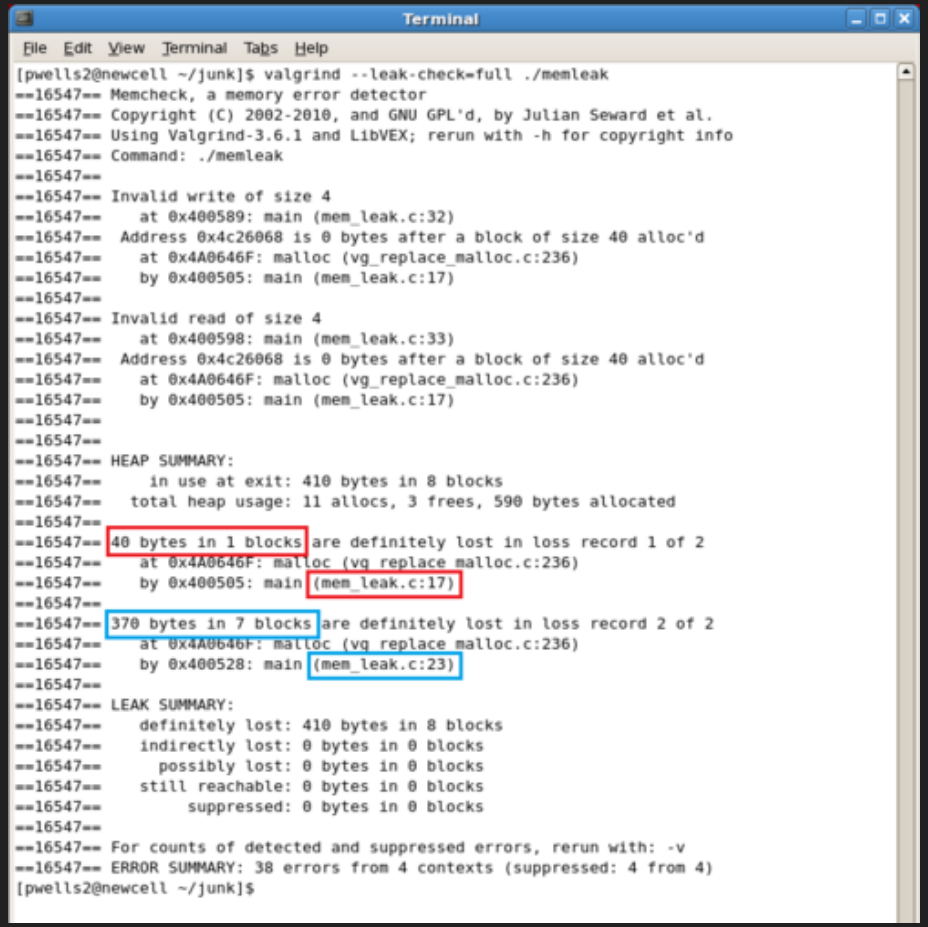
GDB - Examples of commands

gdb ./<executable>	debug "program" (from the shell)
r	run the loaded program with the parameters
bt	backtrace (in case the program crashed)
b	Set Breakpoint
n	Execute the next line
p	Print the value of variable
Info locals	Prints values of variables in scope
q	Quit GDB

Valgrind

Valgrind is an instrumentation framework for building dynamic analysis tools. There are Valgrind tools that can automatically detect many memory management and threading bugs, and profile your programs in detail. You can also use Valgrind to build new tools.

valgrind --tool=memcheck --leak-check=full ./<executable>

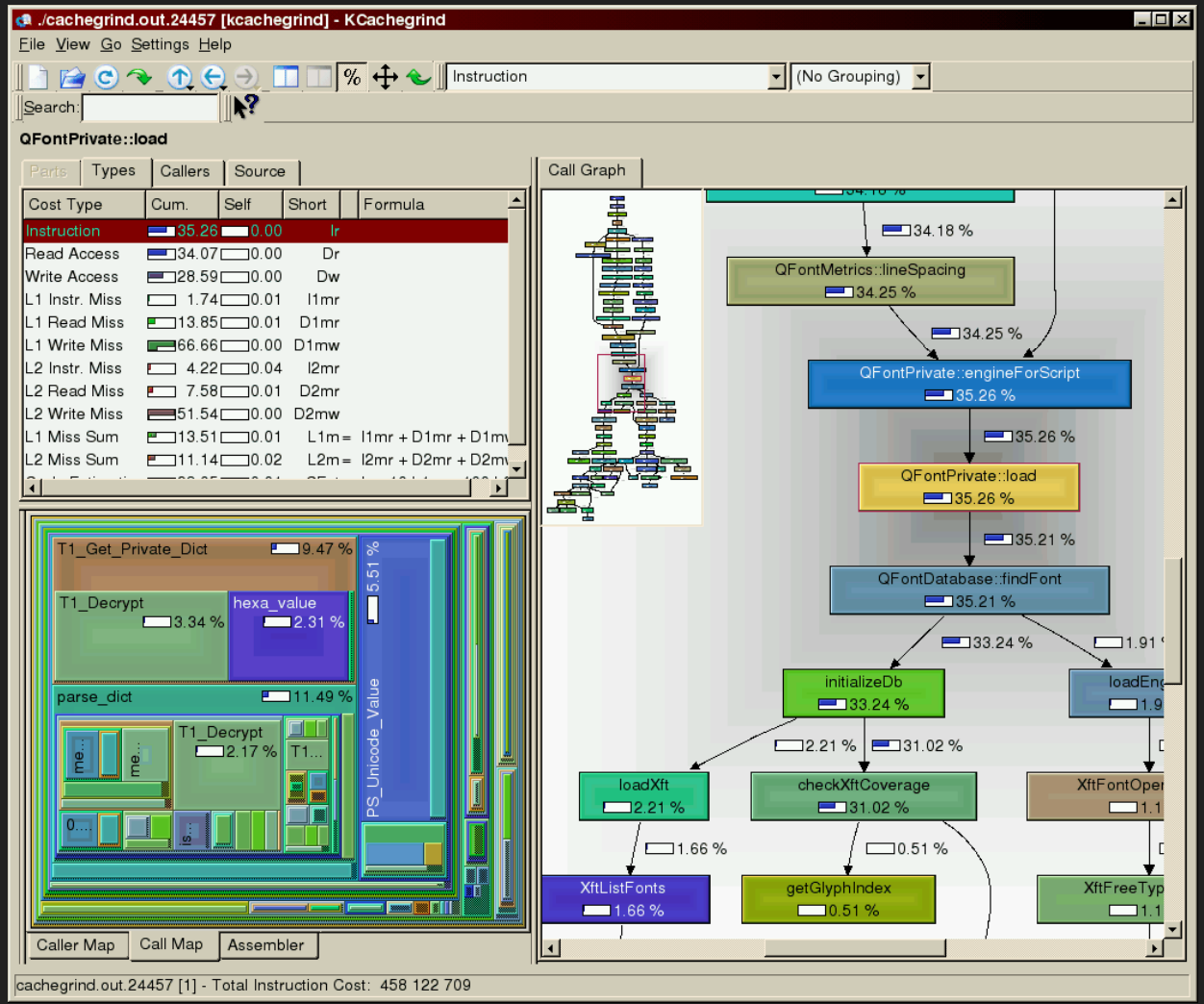


Profiling

In software engineering, profiling ("program profiling", "software profiling") is a form of dynamic program analysis that measures, for example, the space (memory) or time complexity of a program, the usage of particular instructions, or the frequency and duration of function calls. Most commonly, profiling information serves to aid program optimization.

Callgrind and KCACHEGRIND

Callgrind uses runtime instrumentation via the Valgrind framework for its cache simulation and call-graph generation. This way, even shared libraries and dynamically opened plugins can be profiled. The data files generated by Callgrind can be loaded into **KCachegrind** for browsing the performance results. But there is also a command line tool in the package to get ASCII reports from data files without the need to use KCachegrind.



valgrind --tool=callgrind ./<executable>

kcachegrind callgrind.out.8117

Optional:

https://www.youtube.com/watch?v=y5JmQItfFck

https://www.youtube.com/watch?v=fvTsFjDuag8