FERRAMENTA DE PROFILING: Valgrind + Kcachegrind

Por: Gustavo Xavier Pereira

Descrição

A ferramenta de profiling escolhida foi o valgrind, no qual funciona juntamente com o kcachegrind. Valgrind é um software livre que auxilia o trabalho de depuração de programas para as linguagens de programação C e C++. O diferencial deste programa está no fato de que usa uma máquina virtual para simular o acesso à memória do programa em teste, eliminando a necessidade de uso de outras bibliotecas auxiliares ou mudanças drásticas no código. Assim, gerando um relatorio mais "limpo" do código.

Características

O Valgrind suporta uma grande variedade de verificações que podem não ser cobertas pelo próprio compilador. É amplamente capaz de detectar bottlenecks e o número de instruções executadas pela máquina, além de possuir ferramentas que detectam erros decorrentes do uso incorreto da memória dinâmica, como por exemplo os vazamentos de memória, alocação e desalocação incorretas e acessos a áreas inválidas.

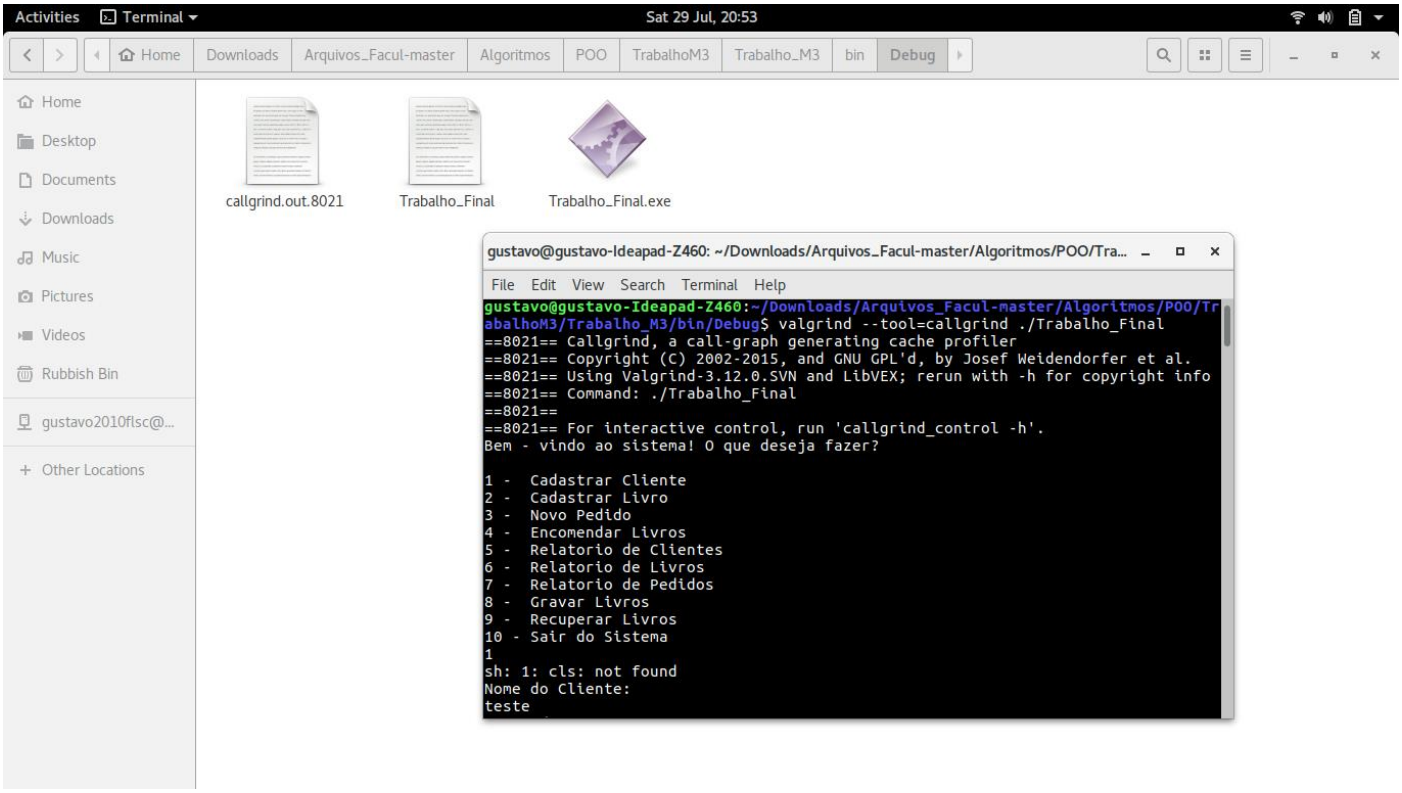
Modo de uso

Após instalar a ferramenta, é preciso executar o programa dentro da máquina virtual do valgrind, no qual gerará um arquivo log, que será lido e interpretado pelo kcachegrind. É recomendável utilizar todas as funcionalidades do programa para uma checagem completa.

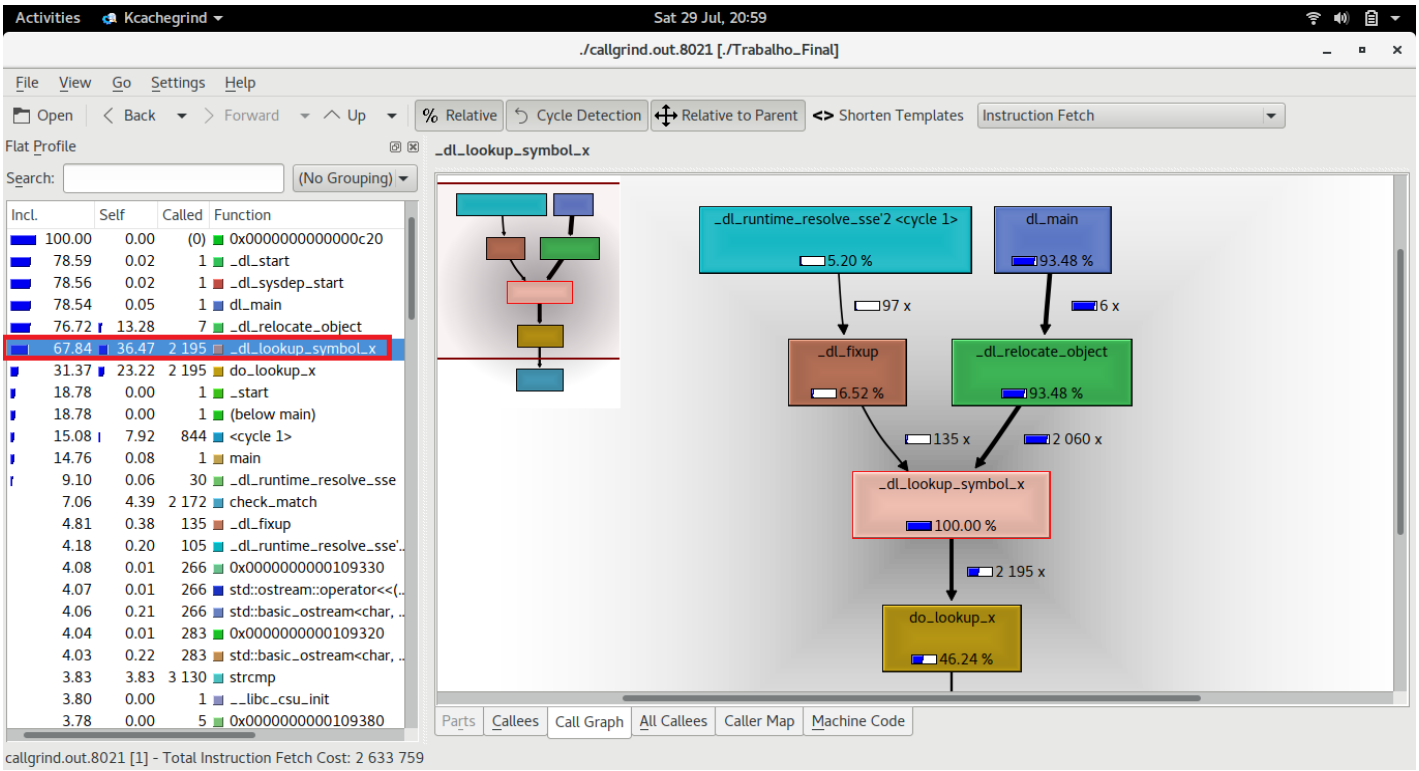
TESTES

Realizei um teste da ferramenta usando um programa gerenciador de vendas de uma livraria, no qual foi o trabalho de algoritmos no semestre passado, utilizando o paradigma da programação orientada a objetos. O programa realiza cadastro de clientes, de livros, gerencia pedidos, vendas, estoque, gravação e recuperação de livros em arquivo externo, etc.

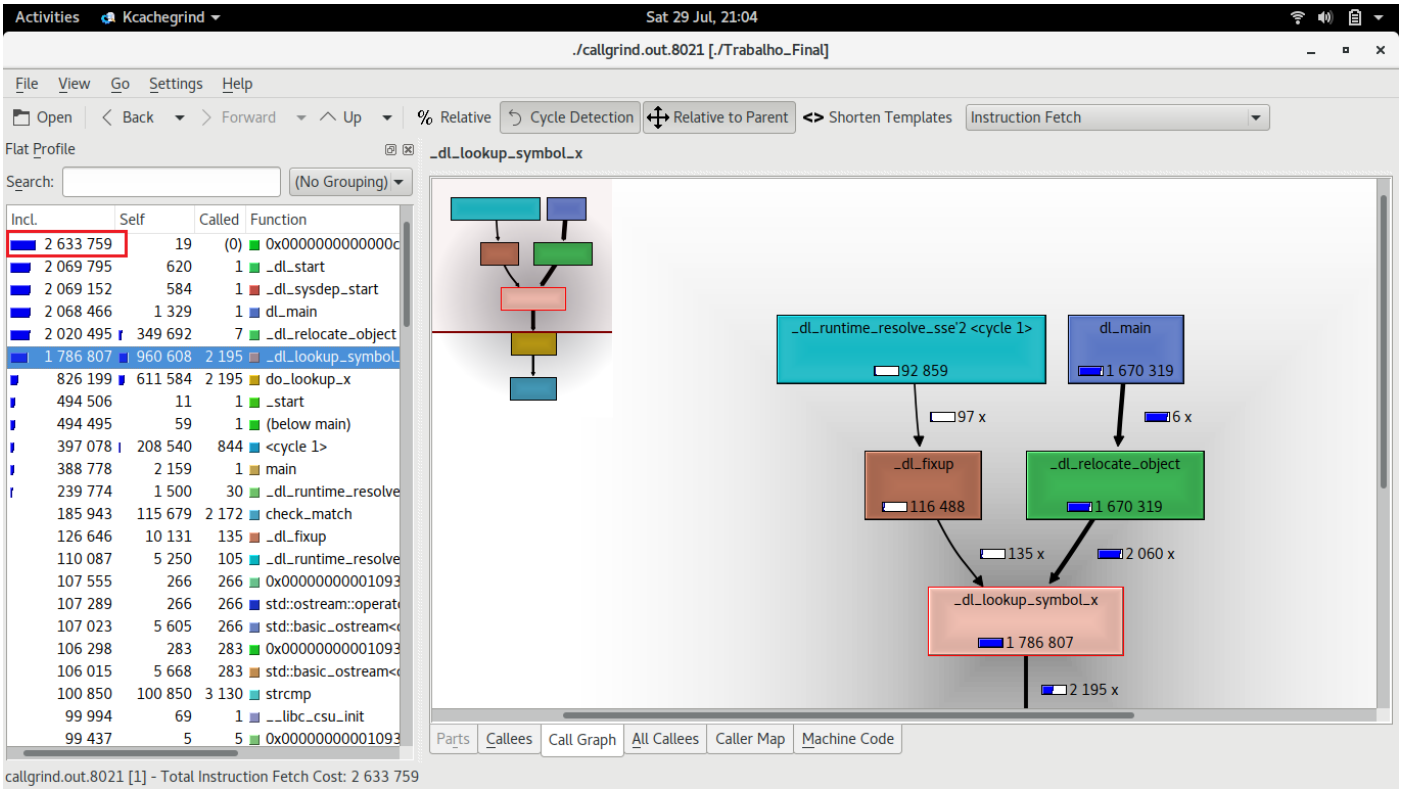
Primeiramente, deve-se executar um scaneamento do programa dentro da interface do valgrind, abaixo uma screenshot:



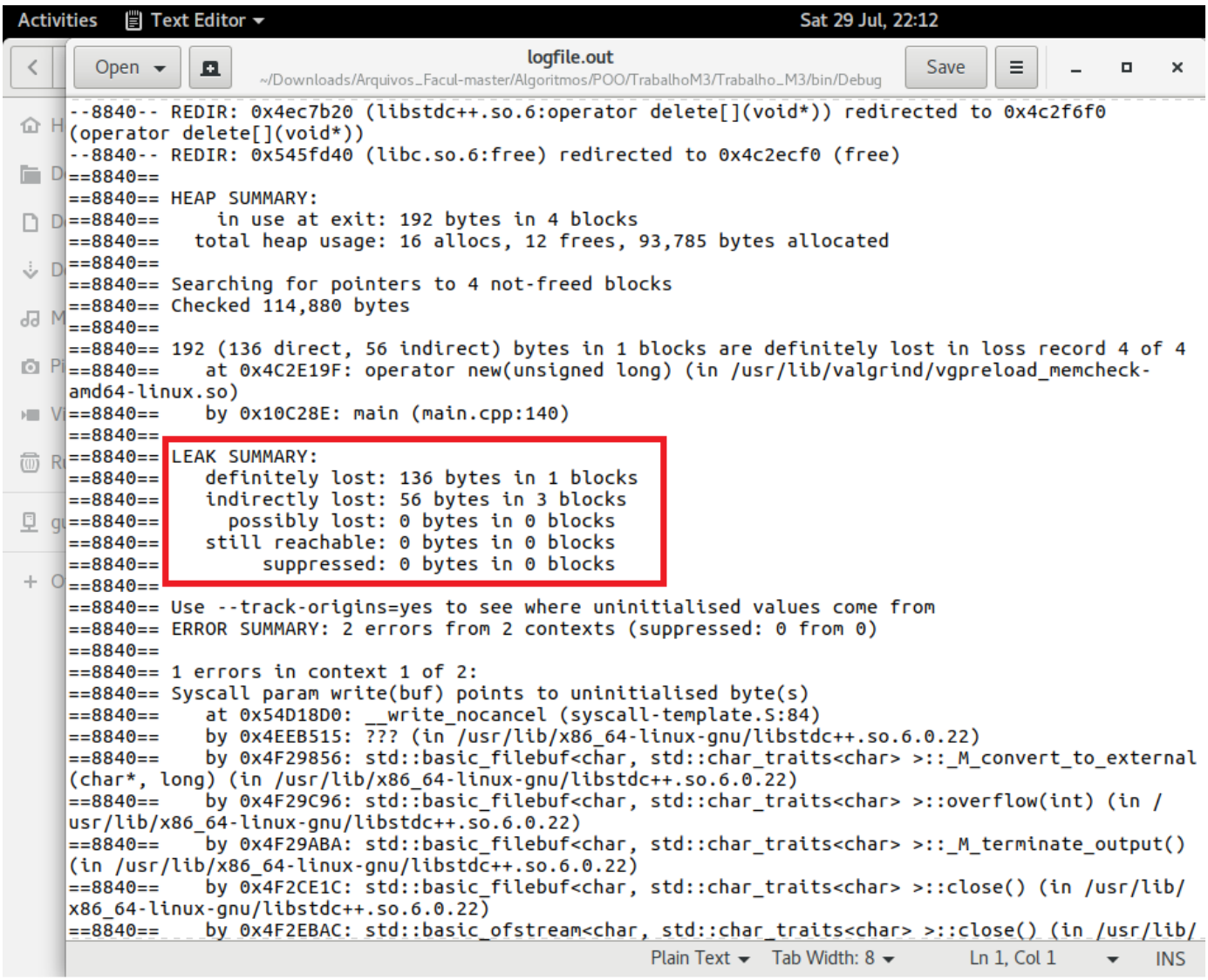
Note que o programa gerou um arquivo log na pasta com o nome “callgrind.out.8021”, que será aberto posteriormente com o kcachegrind.



Na imagem acima, com o log já aberto no kcachegrind, podemos finalmente estudar o código. Note que o kcachegrind exibe uma árvore de funções e seus respectivos consumos. A função selecionada é considerada o “bottleneck”, pois é a função com maior consumo em todo o código. Sozinha está ocupando 36,47% das instruções.



Aqui, desabilitando a opção “Relative”, podemos observar o consumo de instruções de máquina das funções. Note que o programa inteiro utilizou 2 633 759 instruções.



A imagem acima exibe o log do memcheck, no qual mostrou que há vazamentos de memória no arquivo main.cpp e necessita de atenção.

Conclusão

O Valgrind é uma ferramenta extremamente poderosa para o profiling avançado, possui diversas funcionalidades e permite ao programador total conhecimento sobre o que está acontecendo no seu código. Um ponto fraco seria a usabilidade, pois é uma ferramenta complexa e requer um pouco de maestria para ser executada, mas nada que algumas horas “fuçando” não resolvam!

Referências

http://valgrind.org/ - Página oficial do Valgrind

https://pt.wikipedia.org/wiki/Valgrind - Valgrind Wiki

http://kcachegrind.sourceforge.net/html/Home.html - Página na web do Kcachegrind