Profiling (Perfilamento)

Programação Paralela

Mário Douglas Alves Cabral

Bacharelado Ciência da Computação
Instituto Federal Campus Rio Verde

27 de agosto de 2019

Profiling

- Na engenharia de software, profiling ou perfilamento, é uma forma de análise dinâmica de software.
- Tem o objetivo de analisar o desempenho de um programa, por exemplo, mede a memória utilizada, quantidade de uso das instruções.

Criação de perfil

- É obtida através da instrumentação do código fonte ou binário do programa, utilizando a ferramenta chamada profilers (perfiladores).
- Várias técnicas são utilizadas para o perfilamento, variando para cada profiler. Algumas metodologias são baseadas em eventos, estatística, instrumentação e simulação.

Tipos de Perfilador com base na saída

- Perfilador plano;
- Perfilador de gráfico de chamadas;
- Perfilador sensível à entrada

Profilers (Perfilador)

- **Gprof**;
- **Linux Perf**;
- **Windows Performance Toolkit**;
- **Intel VTune Amplifier**;
- **cProfile**;
- **profile**;
- **HPROF**.

Gprof

- Uma ferramenta do projeto GNU, faz parte GNU Binutils, aponta quanto tempo está sendo gasto em cada parte do seu programa.
- Para criação do perfil é necessário:
  - Compilar e vincular o programa com criação de perfil ativada.
    gcc -o myprog.out myprog.c -pg
  - Executar o programa para gerar um arquivo de dados do perfil.
    ./myprog.out
  - Executar gprof para analisar os dados do perfil.
    gprof options [executable-file [profile-data-files...]] [> outfile]

gprof2dot

- Script Python para converter saídas de vários profilers em um gráfico em formato dot.
- Os passos geração são:
  - gprof ./myprog.out gmon.out > profiling.txt
  - ./gprof2dot.py profiling.txt > profiling.dot
  - dot -Tpng -o profiling.png profiling.dot

Flat Profile

```
Flat profile:

Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls   s/call   s/call  name
 46.33      1.18     1.18 100000000     0.00     0.00  retiraInicio_sem_ajustes(filaLinear*)
 24.34      1.80     0.62        1     0.62     0.75  fila_prenche_aletorio(filaLinear*)
 16.88      2.23     0.43 200000001     0.00     0.00  fila_vazia(filaLinear*)
  4.91      2.36     0.13 100000000     0.00     0.00  fila_cheia(filaLinear*)
  4.71      2.48     0.12        1     0.12     1.73  esvaziar(filaLinear*)
  1.77      2.52     0.05                             enfileirar(filaLinear*, int)
  1.18      2.55     0.03        1     0.03     0.03  inicializaFilaVazia(filaLinear*)
  0.00      2.55     0.00        3     0.00     0.00  select_op()
  0.00      2.55     0.00        1     0.00     0.00  _GLOBAL__sub_I_main
  0.00      2.55     0.00        1     0.00     0.00  __static_initialization_and_destruction_0(int, int)
  0.00      2.55     0.00        1     0.00     0.00  criarFila()
```

Call Graph

```
                     Call graph (explanation follows)


granularity: each sample hit covers 2 byte(s) for 0.39% of 2.55 seconds

index % time    self  children    called     name
                                                 <spontaneous>
[1]     98.2    0.00    2.51                 main [1]
                0.12    1.61       1/1           esvaziar(filaLinear*) [2]
                0.62    0.13       1/1           fila_prenche_aletorio(filaLinear*) [4]
                0.03    0.00       1/1           inicializaFilaVazia(filaLinear*) [8]
                0.00    0.00       3/3           select_op() [15]
                0.00    0.00       1/1           criarFila() [18]
-----------------------------------------------
                0.12    1.61       1/1           main [1]
[2]     67.8    0.12    1.61       1         esvaziar(filaLinear*) [2]
                1.18    0.22 100000000/100000000     retiraInicio_sem_ajustes(filaLinear*) [3]
                0.22    0.00 100000001/200000001     fila_vazia(filaLinear*) [5]
-----------------------------------------------
                1.18    0.22 100000000/100000000     esvaziar(filaLinear*) [2]
[3]     54.7    1.18    0.22 100000000         retiraInicio_sem_ajustes(filaLinear*) [3]
                0.22    0.00 100000000/200000001     fila_vazia(filaLinear*) [5]
-----------------------------------------------
                0.62    0.13       1/1           main [1]
[4]     29.2    0.62    0.13       1         fila_prenche_aletorio(filaLinear*) [4]
                0.13    0.00 100000000/100000000     fila_cheia(filaLinear*) [6]
-----------------------------------------------
                0.22    0.00 100000000/200000001     retiraInicio_sem_ajustes(filaLinear*) [3]
                0.22    0.00 100000001/200000001     esvaziar(filaLinear*) [2]
[5]     16.9    0.43    0.00 200000001         fila_vazia(filaLinear*) [5]
-----------------------------------------------
                0.13    0.00 100000000/100000000     fila_prenche_aletorio(filaLinear*) [4]
[6]      4.9    0.13    0.00 100000000         fila_cheia(filaLinear*) [6]
-----------------------------------------------
```

Gráfico Fila Linear