

## Atividade Prática 6: Análise qualitativa do código da atividade prática 4

Diogo Tuler Chaves - 2022043663

### 1. Passo 1

O código da atividade prática 4 foi a implementação das operações criapilha, pilhavazia, empilha, desempilha e destroiapilha utilizando uma ou mais estruturas de dados fila circular. Em outras palavras, o código foi a construção de uma Pilha a partir de filas circulares.

Todas as alocações são feitas pela fila, logo a análise vai ser feita em volta de suas funções. A função mais crítica é a empilha, que cria elementos "Fila\_Circular" alocados dinamicamente. Na minha visão, esse código pode ter dois comportamentos, dependendo de como essa função é chamada. Se ela for chamada várias vezes seguidas, provavelmente cada elemento vai ser alocado relativamente próximo na memória, pois o programa deve entender que, por terem sido armazenadas em um período de tempo curto, provavelmente vão ser acessadas juntas em um período próximo. Todavia, caso o programa chame a função empilha em momentos variados com bastante tempo entre as alocações o programa não deve entender que esses elementos vão podem ser acessados juntos, logo eles podem ser armazenados em lugares variados.

O armazenamento mais disperso vai ser negativo para o programa, pois ele pode ter que acessar a RAM para achar elementos armazenados, pois eles podem não estar nas mesmas vagas de memória escolhidas para serem os cachês, aumentando o tempo gasto pelo programa.

### 2. Passo 2

O código vai ser executado com Cachegrind e Callgrind para uma análise de miss cache que vai acontecer. Isso vai acontecer pois misses caches muitos altos indicam que o programa não está utilizando do princípio da localidade de forma efetiva, tendo que acessar níveis mais lentos de armazenamentos para acessar o necessário para rodar o código. Ambas essas ferramentas foram escolhidas pois foram as recomendadas no material da aula.

### 3. Passo 3

```
#define LIMITE 10000
```

Essa variável constante define a quantidade de valores a serem armazenados e o range desses valores, valendo ressaltar que cada um é único.

### 4. Passo 4

```

I1 cache: 32768 B, 64 B, 8-way associative
D1 cache: 32768 B, 64 B, 8-way associative
L1 cache: 6291456 B, 64 B, 12-way associative
Command: ./bin/main
Data file: cachegrind.out.167434
Events recorded: Ir Ilnr ILnr Dr Dlnr DLnr Dw Dlnw DLnw
Events shown: Ir Ilnr ILnr Dr Dlnr DLnr Dw Dlnw DLnw
Event sort order: Ir Ilnr ILnr Dr Dlnr DLnr Dw Dlnw DLnw
Thresholds: 0.1 100 100 100 100 100 100 100
Include dirs:
User annotated:
Auto-annotation: on
-----
Ir Ilnr ILnr Dr Dlnr DLnr Dw Dlnw DLnw
28,066,512,007 (100.0%) 1,369 (100.0%) 1,355 (100.0%) 8,534,833,544 (100.0%) 14,602,697 (100.0%) 1,364 (100.0%) 2,905,166,772 (100.0%) 50,155,966 (100.0%) 11,396 (100.0%) PROGRAM TOTALS
-----
Ir Ilnr ILnr Dr Dlnr DLnr Dw Dlnw DLnw file:function
2,265,300,512 (26.32%) 3 ( 0.22%) 3 ( 0.22%) 2,826,140,774 (33.11%) 13,957,529 (95.58%) 0 350,168 ( 0.01%) 0 0 ????:Ja_Foi
1,199,890,002 (15.99%) 8 ( 0.58%) 8 ( 0.59%) 999,940,016 (11.72%) 9,998 ( 0.07%) 0 549,935,028 (18.93%) 0 0 ./malloc./malloc/malloc.c:_int_f
ree
2,550,145,000 (12.75%) 4 ( 0.29%) 4 ( 0.30%) 1,600,060,000 (18.75%) 0 0 450,015,000 (15.49%) 49,985,001 (99.60%) 0 ????:Desenfileira
2,150,235,124 (10.75%) 7 ( 0.51%) 7 ( 0.52%) 550,055,032 ( 6.44%) 60,001 ( 0.41%) 0 299,990,014 (10.33%) 1 ( 0.00%) 0 ./malloc./malloc/malloc.c:malloc
1,700,060,000 ( 8.50%) 2 ( 0.15%) 2 ( 0.15%) 750,015,000 ( 8.79%) 0 0 550,015,000 (18.93%) 0 0 ????:Enfileira
1,650,375,048 ( 8.25%) 1 ( 0.07%) 1 ( 0.07%) 600,140,018 ( 7.03%) 9,998 ( 0.07%) 0 300,050,006 (10.33%) 0 0 ????:Fila_Vazia
1,099,890,044 ( 5.50%) 3 ( 0.22%) 3 ( 0.22%) 399,960,016 ( 4.69%) 9,999 ( 0.07%) 0 199,980,008 ( 6.88%) 0 0 ./malloc./malloc/malloc.c:free
750,264,988 ( 3.75%) 2 ( 0.15%) 2 ( 0.15%) 250,084,997 ( 2.93%) 0 0 150,074,996 ( 5.17%) 0 0 ????:Empilha
700,070,000 ( 3.50%) 1 ( 0.07%) 1 ( 0.07%) 250,025,000 ( 2.93%) 0 0 250,025,000 ( 8.61%) 0 0 ????:Cria_Fila
550,165,022 ( 2.75%) 1 ( 0.07%) 1 ( 0.07%) 200,060,008 ( 2.34%) 10,000 ( 0.07%) 0 150,045,006 ( 5.16%) 10,000 ( 0.02%) 0 ????:Pilha_Vazia
200,547,002 ( 1.00%) 29 ( 2.12%) 28 ( 2.07%) 100,262,606 ( 1.17%) 56,029 ( 0.38%) 0 12 ( 0.00%) 1 ( 0.00%) 1 ( 0.01%) ????:???
149,985,000 ( 0.75%) 0 0 0 0 0 0 0 0 0 ./malloc./malloc/arena.c:free
-----
The following files chosen for auto-annotation could not be found:

```

## 5. Passo 5

```

I1 cache:
D1 cache:
L1 cache:
Timerange: Basic block 0 - 3922420195
Trigger: Program termination
Profiled target: ./bin/main (PID 167899, part 1)
Events recorded: Ir
Events shown: Ir
Event sort order: Ir
Thresholds: 99
Include dirs:
User annotated:
Auto-annotation: on
-----
Ir
-----
21,861,041,901 (100.0%) PROGRAM TOTALS
-----
Ir file:function
7,117,337,768 (32.56%) ????:Ja_Foi [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
3,199,890,000 (14.64%) ./malloc./malloc/malloc.c:_int_free [/usr/lib/x86_64-linux-gnu/libc.so.6]
2,550,145,000 (11.67%) ????:Desenfileira [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
2,150,235,124 ( 9.84%) ./malloc./malloc/malloc.c:malloc [/usr/lib/x86_64-linux-gnu/libc.so.6]
1,700,060,000 ( 7.78%) ????:Enfileira [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
1,650,375,048 ( 7.55%) ????:Fila_Vazia [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
1,099,890,044 ( 5.03%) ./malloc./malloc/malloc.c:free [/usr/lib/x86_64-linux-gnu/libc.so.6]
750,264,988 ( 3.43%) ????:Empilha [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
700,070,000 ( 3.20%) ????:Cria_Fila [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
550,165,022 ( 2.52%) ????:Pilha_Vazia [/home/diogo/Documentos/Faculdade/Códigos/Materias/Periodo3/ESTRUTURA_DE_DADOS/aulapratica4/bin/main]
149,985,006 ( 0.69%) ./malloc./malloc/arena.c:free
100,010,004 ( 0.46%) ????:0x000000000109150 [???]
-----
The following files chosen for auto-annotation could not be found:

```

## 6. Passo 6

### a. Quão bem o programa se comporta em termos de memória?

Como podemos ver, a maioria das instruções executadas provém da função “Já Foi”, que no código é usada para verificar se o número escolhido já foi empilhado ou não, tento de ser usada até se encontrar um número que não foi ainda. Em relação ao cache miss, a maioria ocorreu nessa função

também sendo um valor bem alto, tendo D1mr ( que seria o número de acesso a dados que resultam em miss cache no primeiro nível de dados) de 13,957,529 no total de 2,826,140,774 acessos totais. Esse valor não é muito bom, mas também não é tão ruim, tendo em vista que essa quantidade de D1mr representam 95% dos D1mr mas apenas 0,5% dos D1 dessa função apenas. Logo em termos de memória esse código se comporta muito melhor que o esperado, todavia ainda são necessários ajustes para uma maior utilização da localidade de referência.

**b. Quais estruturas de dados devem serem caracterizadas para melhor entendimento?**

As estruturas mais acessadas são o Ir cache (representa as instruções a serem realizadas) e o Dr cache (representa o número total de acessos a dados para leitura ), com um adendo ao DW cache (representa o número total de acessos a dados para escrita). Todos esses tiveram índices de acesso altos e de caches misses mais controlados, tirando o DW em alguns casos, todavia poderiam ser otimizados.

**c. Quais segmentos de código devem instrumentados para suportar a caracterização?**

Obviamente é a função “Já Foi” que acumula 35% do Dr Cache e 26% da Ir Cache. Além disso, ela acumula 95% dos D1mr, a sua instrumentação traria uma análise mais precisa sobre seu impacto no desempenho e poderá identificar áreas que requerem otimizações. A segunda função mais usada é o malloc dentro dessa função, tornando cada vez mais evidente a necessidade de sua análise específica.