

Aberto: domingo, 8 out. 2023, 13:30

Vencimento: quarta-feira, 29 nov. 2023, 23:59

Trabalho 02: Otimização de Ajuste Polinomial de Curvas com Cálculo Intervalar

O objetivo deste trabalho é melhorar e avaliar o desempenho do **Ajuste Polinomial com Cálculo Intervalar** desenvolvido no **Trabalho 01**.

Melhoria de Desempenho

Você deve alterar o código do primeiro trabalho (**v1**) de forma a obter uma melhora no desempenho (**v2**).

As alterações no Trabalho 01, seja na geração do SL, na solução do SL, nas estruturas de dados ou qualquer outro ponto relevante, devem ser **explicadas** no relatório a ser entregue, **justificando** as razões pelas quais você efetuou cada alteração.

Análise de Desempenho

Uma vez alterado o código, você deve comparar o desempenho das duas versões em relação a: **(A)** geração do SL pelo Método dos Mínimos Quadrados; **(B)** solução do SL pelo Método da Eliminação de Gauss, e **(C)** cálculo do resíduo (incluindo o cálculo de $f(\mathbf{x}_i)$). Estas análises devem ser descritas sob a forma de um **trabalho acadêmico**, e entregues em formato PDF.

É imprescindível que sejam respeitadas as seguintes condições:

1. Ambos códigos devem ser compilados com GCC e as opções: **-O3 -mavx -march=native**;
2. Os códigos devem ser compilados na mesma máquina utilizada para os testes;
3. Os testes devem utilizar os mesmos parâmetros e em igualdade de condições;
4. Ambos códigos devem ser instrumentados com a biblioteca do **LIKWID** e executados na **core** mais superior (no caso do DINF, na **core 3**). Use a opção **-C <core>** ao executar o programa via **likwid-perfctr**;
5. Você pode escolher um computador de sua preferência, desde que possua os contadores Likwid especificados. Não utilize as servidoras de processamento do DINF que tenham uso compartilhado. Elas podem ser máquinas virtuais e o compartilhamento impede medidas de desempenho. **Em caso de dúvida, consulte o professor**;
6. Você deve apresentar a arquitetura do processador utilizado nos testes no seu texto. Estas informações podem ser obtidas com o comando **"LIKWID-topology -g -c"**.

Para comparar o desempenho dos códigos, você deve efetuar uma série de testes.

?

- **Cada teste** deve ser reportado sob a forma de **três** gráficos de linhas, um para o item **(A)** (geração do SL pelo Método dos Mínimos Quadrados), um para o item **(B)** (solução do SL pelo Método da Eliminação de Gauss), e outro para o item **(C)** (cálculo de resíduo);
- **Cada gráfico** deve ter linhas em cores distintas para os resultados de cada versão (**v1** e **v2**);
- No eixo das **abscissas** os gráficos representam o número de pontos **K**. Cada teste deve ser executado para os seguintes valores de **K**: 64, 128, 200, 256, 512, 600, 800, 1024, 2000, 3000, 4096, 6000, 7000, 10000, 50000, 10^5 , 10^6 , 10^7 , 10^8 ;
 - Para gerar os dados de teste, compile o programa **gera_entrada** que gera **K** pontos para um ajuste de curva polinomial de grau 4 (vide Arquivos auxiliares abaixo) e use-o em pipeline com seu programa (onde **<Kpontos>** é a quantidade de pontos a ser gerado):

```
./gera_entrada <Kpontos> | ./ajustePol (sem LIKWID)
```

```
./gera_entrada <Kpontos> | likwid-perfctr -C 3 -g <metrica> -m ./ajustePol (com LIKWID)
```

Por exemplo, para testar o ajuste para 100 pontos, execute:

```
./gera_entrada 100 | ./ajustePol
```

- **ATENÇÃO:** em função dos tamanhos muito grandes de **K**, algumas variáveis usadas no programa **ajustePol** deverão ser do tipo **long long int**. Essa modificação deve ser feita em ambas versões (**v1** e **v2**), a fim de permitir que os testes sejam realizados.
- **Cada gráfico** deve ser explicado e você deve demonstrar que consegue entender o que está reportado nele;

- Os gráficos devem ser apresentados com o eixo das abscissas em **escala logarítmica**.

Os seguintes testes devem ser executados:

1. **Teste de tempo:** mostrar o tempo de execução de cada item monitorado usando `timestamp()`. No gráfico, esta **ordenada** deve estar também em **escala logarítmica**;
2. **Operações aritméticas:** utilizar o grupo FLOPS_DP do LIKWID e reportar **FLOPS DP** e **FLOPS AVX DP**, em **MFLOP/s**. No Relatório, deve ser explicado os resultados obtidos em FLOPS AVX DP;
3. **Banda de Memória:** utilizar o grupo MEM do LIKWID, e apresentar o resultado de "Memory bandwidth [MBytes/s]". Caso não tenha o grupo MEM, utilize o grupo L3;
4. **Cache miss L2:** utilizar o grupo CACHE do LIKWID, e apresentar o resultado de "**data cache miss RATIO**". Caso não tenha o grupo CACHE, utilize o grupo L2CACHE. Caso não tenha o cache miss da L2, utilize o cache miss da L3 (L3CACHE).

ATENÇÃO: para os itens **(A)** e **(C)** (geração do SL e cálculo de resíduo) devem ser gerados os gráficos de todos os testes acima. Para o item **(B)** (solução do SL) devem ser gerados os gráficos **APENAS** para os testes de **Tempo** e **Operações aritméticas**.

Produto a ser entregue

O trabalho deve ser desenvolvido pelo **MESMO GRUPO** da primeira parte.

Cada grupo deve entregar um pacote de software completo contendo os fontes em **linguagem C** do trabalho 1 (**v1**), com as modificações introduzidas para o trabalho 2 (**v2**) e um relatório em formato PDF.

Cada grupo deve entregar via MOODLE C3SL um pacote de software completo contendo os fontes em linguagem C, arquivos LEIAME e Makefile. O pacote deve ser arquivado e compactado com **zip** ou **tar**, em um arquivo chamado **login1.<ext>** (se grupo com 1 membro) ou **login1-login2.<ext>** (se grupo com 2 membros), onde **login1** e **login2** são os logins (nos sistemas do DINF) dos alunos que compõem o grupo. O pacote deve ter a mesma estrutura de diretório e arquivos do Trabalho 1. Lembre-se que **<ext>** é **tar**, **tar.gz** ou **zip**.

O pacote deve ter a seguinte estrutura de diretório e arquivos:

- **./login1-login2/**: diretório principal
- **./login1-login2/LEIAME**: arquivo contendo descrição geral dos módulos e qualquer outro esclarecimento sobre o programa.
- **./login1-login2/RELATORIO-login1-login2.pdf**: arquivo contendo o relatório do trabalho
- **./login1-login2/Makefile**
- **./login1-login2/*.c**
- **./login1-login2/*.h**

Note que a extração dos arquivos de **login1-login2.<ext>** deve criar o diretório **login1-login2** contendo todos os arquivos acima. Os arquivos fonte também devem estar contidos no diretório, ou em algum sub-diretório, desde que o **Makefile** funcione.

O relatório deve ter o formato de um trabalho acadêmico e conter todas as informações e análises solicitadas. Além disso, a conclusão deve apresentar os aspectos que sua equipe considerou mais relevante/importante no desenvolvimento do trabalho.

Entrega

O prazo final para a entrega deste trabalho é dia **29 de novembro de 2023, 23:59:00h**, IMPRETERIVELMENTE.

- O trabalho deve ser entregue via moodle

Critérios de Avaliação

APENAS OS TRABALHOS QUE FUNCIONAREM SERÃO CORRIGIDOS. Se o trabalho não compilar ou acusar falha de segmentação (Segmentation fault) prematura durante os testes realizados pelo professor (sem que qualquer operação se efetue a contento) ou não produzir uma solução correta, trará para o grupo NOTA 0 (ZERO). Também receberão NOTA 0 (ZERO) trabalhos plagiados de qualquer fonte, e/ou com códigos idênticos ou similares. Além disso, apenas trabalhos entregues no prazo marcado receberão nota.

Os itens de avaliação do trabalho e respectivas pontuações são:

- Qualidade da documentação: arquivo RELATORIO-login1-login2.pdf (20 pontos)
- Testes: correteude das respostas nos testes executados bem como na sua análise (40 pontos)
- Otimização: Qualidade/dificuldade das otimizações efetuadas (40 pontos)

Arquivos auxiliares

Para implementação de funções auxiliares (como `timestamp()`) e scripts para geração de gráficos, pode-se usar o git abaixo (pastas `utils` e `gnuplot`):

<https://gitlab.c3sl.ufpr.br/nicolui/ci1164-utils>

Para geração de dados de entrada, compile e use o programa `gera_entrada.c` abaixo conforme indicado anteriormente. Para compilar use:

```
gcc gera_entrada.c -o gera_entrada
```

 [gera_entrada.c](#)

9 outubro 2023, 13:29 PM

Adicionar envio

Status de envio

Número da tentativa	Esta é a tentativa 1 .
Status de envio	Nenhum envio foi feito ainda
Status da avaliação	Não há notas
Tempo restante	48 dias 11 horas restando
Última modificação	-
Comentários sobre o envio	<div>▶ Comentários (0)</div>