EP2: Criptografia em GPUs usando CUDA

Pedro Bruel e Alfredo Goldman

MAC 5742-0219 Introdução à Programação Concorrente, Paralela e Distribuída

1. Introdução

Neste EP vocês implementarão versões para GPU, usando CUDA, de alguns algoritmos simples de encriptação e geração de *hashes* de arquivos. Forneceremos implementações em `C` dos algoritmos, e cada grupo deverá implementar no mínimo 3 algoritmos usando CUDA.

O código na linguagem `C` e os arquivos em LaTeX necessários para gerar este documento estão disponíveis no *GitHub*[1]. O resto deste documento descreve as tarefas que você e seu grupo deverão realizar no EP2.

2. Tarefas

Vocês devem implementar em CUDA no mínimo 3 dos algoritmos de encriptação ou *hash* disponibilizados no repositório do EP. Vocês podem usar 3 algoritmos de encriptação ou *hash* de outras fontes, **desde que entreguem também o código sequencial do algoritmo**.

Depois, devem realizar experimentos comparando o desempenho das versões sequenciais e em CUDA, usando os arquivos de teste disponibilizados no repositório, ou arquivos escolhidos por vocês. Caso usem arquivos diferentes dos disponibilizados, incluam *links* no relatório para lugares onde os arquivos estão hospedados.

O arquivo `src/crypto-algorithms/des_test.c` contém uma função para ler e escrever um arquivo como um *stream* de bytes. Vocês podem usar essa função para encriptar arquivos ou escrever a sua própria. Vocês devem escrever **funções de teste** que comparem os arquivos antes da encriptação com os arquivos após uma encriptação e decriptação, ou com o *hash* do arquivo.

Vocês devem entregar um relatório contento gráficos e análises de desempenho dos seus programas e das versões sequenciais. Vocês devem justificar, usando dados e gráficos, as escolhas para tamanho de *block* e *grid* dos seus programas em CUDA.

[1] `https://github.com/phrb/MAC5742-0219-EP2` [Acessado em 11/05/2017]

2.1. Apresentação dos Resultados

Depois de realizar os experimentos vocês deverão elaborar gráficos que evidenciem o comportamento dos três algoritmos com relação à variação do tamanho do arquivo processado. Os gráficos deverão ser claros e legíveis, com eixos nomeados. Deverão apresentar a média e o desvio padrão de 10 execuções de cada programa em cada arquivo.

Recomendamos que vocês usem ferramentas como a biblioteca `matplotlib` da linguagem `Python`. Se fizerem isso vocês conseguirão automatizar a realização dos experimentos e a geração dos gráficos. A automatização dos experimentos e da visualização dos dados gerados é fundamental para pesquisa em Ciência da Computação, pois permite gerar e analisar grandes conjuntos de dados sem muito esforço manual.

2.2. Discussão dos Resultados

Vocês deverão analisar os resultados obtidos e tentar responder a algumas perguntas, para cada um dos 3 algoritmos:

- Como o tempo de execução varia conforme o tamanho do arquivo?
- Como e por quais razões vocês escolheram o tamanho de *block* e *grid*?
- Qual o impacto das operações de I/O e alocação de memória no tempo de execução?

Vocês conseguem pensar em mais perguntas interessantes?

2.3. Entrega no PACA

Vocês deverão entregar no PACA **apenas um relatório e código fonte por grupo**. A entrega deve ser um único arquivo nos formatos `.tar`, `.zip`, ou qualquer formato que o `tar` consiga descompactar. A entrega deve ser feita **até dia 01/06/17**, e deve conter o código em CUDA, código sequencial de algoritmos extras que vocês utilizarem, `makefiles`, código de testes e um relatório no formato `pdf`.

3. Tecnologias

Esta seção descreve brevemente algumas tecnologias usadas no EP2. O monitor estará disponível na **Sala 120** para tirar dúvidas do EP2, envie um e-mail para `pedro.bruel@gmail.com` para marcar um horário.

3.1. Acesso a GPUs da NVIDIA

Vocês podem usar quaisquer GPUs da NVIDIA a que tenham acesso, mas também podemos fornecer acesso às GPUs do nosso laboratório no CCSL para todos que precisarem. Teremos que gerenciar os conflitos de uso, pois só um grupo poderá usar uma dada GPU por vez. Cadastrem seus grupos usando o questionário no PACA e forneçam um e-mail para contato.

3.2. Shell scripting & GNU `screen`

Para deixar os experimentos rodando na sua máquina, faça o seguinte:

```
$ screen
$ ./my_experiments.sh
<Ctrl+A><D>
```

O comando `screen` lança uma seção da qual você pode se desconectar sem parar a execução de um comando. A sequência `<Ctrl+A>` seguida de `<D>` desconectará você da sessão. Para voltar, basta executar:

```
$ screen -r
```

3.3. LaTeX

Instalem o LaTeX na máquina que vão usar para escrever o relatório e usem o arquivo `enunciado_ep2.tex` e o `Makefile` no repositório do EP2 como modelo.

4. Critério de Avaliação

A nota do EP2 vai de **0.0** a **10.0**, e a avaliação será feita da maneira descrita a seguir, se os alunos concordarem.

- Implementação: **6.0**
  - Código compila sem erros e *warnings*: **2.7**
  - Código executa sem erros e produz o resultado correto: **2.7**
  - Boas práticas de programação e clareza do código: **0.6**
- Relatório: **4.0**
  - Apresentação e Análise dos Experimentos: **3.4**
  - Clareza do texto e figuras: **0.6**

O monitor estará disponível na **Sala 120** para tirar dúvidas do EP2, envie um e-mail para `pedro.bruel@gmail.com` para marcar um horário. Bom EP!