

# Programação Paralela

elc-139

## Mandelbrot

## Geração de Fractais em OpenMP

Adriano Luís de Almeida





# Sumário

- Modelo Mandelbrot
- Solução Fractalpar1
  - Schedule static
  - Script de testes
  - Resultados Obtidos
- Solução Fractalpar2
  - Schedule dynamic
  - Shared
  - Resultados Obtidos
- Comparação



# Modelo Mandelbrot

Fractal definido como um conjunto de  $c$  em um plano complexo definido pela sequência:

$$z_0 = 0$$

$$z_{n+1} = z_n^2 + c$$

Para cada **ponto C** a equação se expande e chega formando um novo conjunto infinito de pontos.

$$c = x + iy$$

$$Z_0 = 0$$

$$Z_1 = Z_0^2 + c$$

$$= x + iy$$

$$Z_2 = Z_1^2 + c$$

$$= (x + iy)^2 + x + iy$$

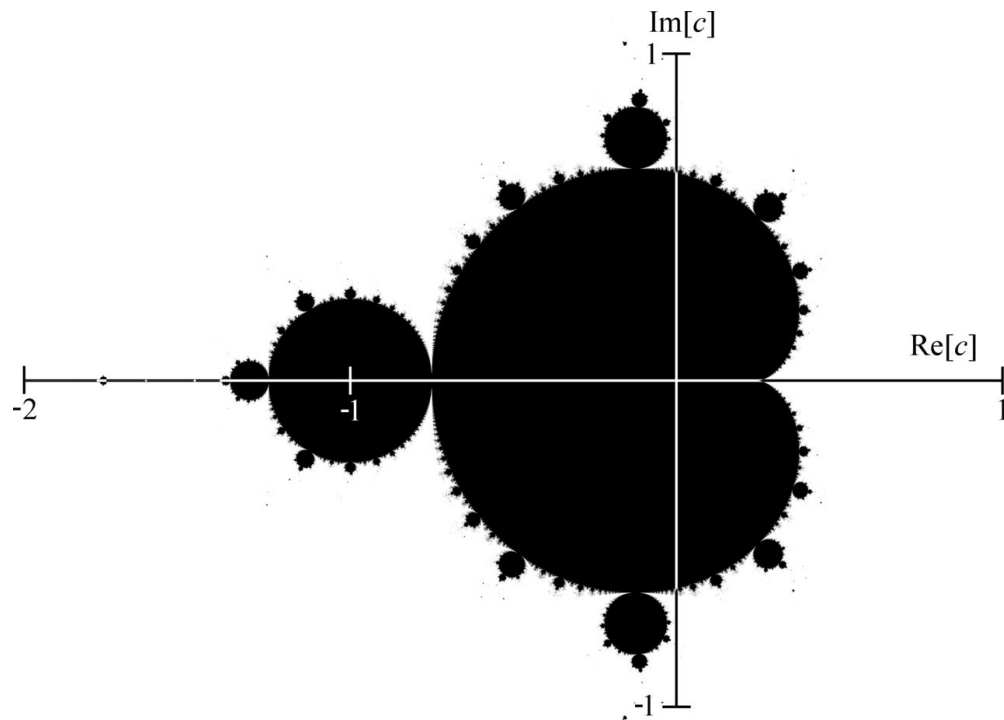
$$= x^2 + 2ixy - y^2 + x + iy$$

$$= x^2 - y^2 + x + (2xy + y)i$$

$$Z_3 = Z_2^2 + c = \dots$$



# Modelo Mandelbrot





```
// compute frames  
double delta = Delta;  
#pragma omp parallel num_threads(nThreads)  
{  
    |  
    #pragma omp for schedule(static, chunk size)
```

# Escalonamento Estático

As iterações do *loop* são divididas em janelas com tamanhos definido pelo *chunk size* e, após, são atribuídas estaticamente para as *threads*.

Chuck size = Frames / n° threads



# Script de Testes

Script desenvolvido usando shell script.

Conjuntos realizam os testes e formam os 3 grandes laços:

```
threads=( 2 4 6 8 )
```

```
width=( 256 512 )
```

```
frames=( 32 64 128 )
```

## Resultados que são realizados nos testes:

- Tempo de execução em Seg.
- Speedup  $T(s) / T(p) \approx 2$
- Eficiência (%)



# Resultados obtidos

Fractalpar1 - Thread(s) 2		
Speedup ( $\approx 2$ ) / Eficiência (%)		
Frames	256	512
32	1.78 / 89%	1.71 / 85 %
64	1.93 / 96%	1.97 / 98%
128	1.81 / 90%	2.0 / 100 %
Fractalpar1 - Thread(s) 4		
Speedup ( $\approx 2$ ) / Eficiência (%)		
Frames	256	512
32	3.21 / 80%	3.02 / 75 %
64	3.05/ 76%	3.16 / 79%
128	3.21 / 80%	3.12 / 78 %

Fractalpar1 - Thread(s) 6		
Speedup ( $\approx 2$ ) / Eficiência (%)		
Frames	256	512
32	3.15 / 52%	3.11 / 51 %
64	3.17/ 52%	3.08 / 51%
128	3.21 / 53%	3.22 / 53 %
Fractalpar1 - Thread(s) 8		
Speedup ( $\approx 2$ ) / Eficiência (%)		
Frames	256	512
32	3.02/ 37%	3.05 / 38 %
64	3.10/ 38%	3.17 / 39%
128	3.23 / 40%	3.78 / 47.28%



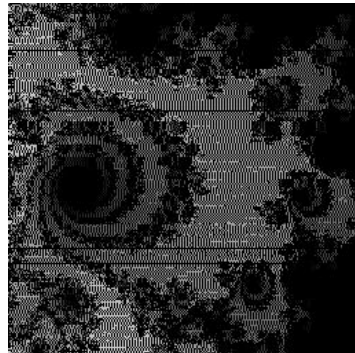
# Solução Fractalpar2 - FAIL

Teste 1 usando escalonamento dinâmico e compartilhamento de recursos.

```
// compute frames
double delta = Delta;
#pragma omp parallel shared(width, frame, row, col) num_threads(nThreads)
{
    #pragma omp for schedule(dynamic)
```

**FAIL**

A cláusula **compartilhada** declara as variáveis na lista a serem compartilhadas entre todos os encadeamentos de uma equipe. Todos os segmentos dentro de uma equipe acessam a mesma área de armazenamento para variáveis compartilhadas.





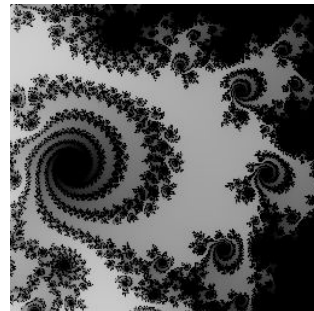


# Solução Fractalpar2 - SUCCESS!

```
// compute frames
double delta = Delta;
#pragma omp parallel private(col, row, frame) shared(pic) num_threads(nThreads)
{
    #pragma omp for schedule(dynamic)
```

A cláusula **privada** declara as variáveis na lista como privadas para cada encadeamento de uma equipe.

## SUCCESS





# Resultados Obtidos - Fractalpar2

## Erro / Sucesso

Fractalpar2 - Thread(s) 2		
Speedup ( $\sim 2$ ) / Eficiência (%)		
Frames	256	512
32	2.59 / 124%	2.97 / 143 %
64	2.89 / 144%	3.31 / 148%
128	2.86 / 143%	3.11 / 155 %
Fractalpar2 - Thread(s) 4		
Speedup ( $\sim 2$ ) / Eficiência (%)		
Frames	256	512
32	5.95 / 148%	6.95 / 173 %
64	5.44 / 136%	6.12 / 153%
128	6.77 / 169%	7.92 / 198 %

Error

|

Fractalpar2 - Thread(s) 2		
Speedup ( $\sim 2$ ) / Eficiência (%)		
Frames	256	512
32	1.96 / 98%	2.02 / 101 %
64	1.69 / 79%	1.96 / 98%
128	2.08 / 104%	2.03 / 101 %
Fractalpar2 - Thread(s) 4		
Speedup ( $\sim 2$ ) / Eficiência (%)		
Frames	256	512
32	2.41 / 60%	2.74 / 69 %
64	2.78 / 69 %	2.85 / 71%
128	3.16 / 79%	2.94 / 73%

Success



# Resultados Obtidos - Fractalpar2

## Erro / Sucesso

Fractalpar2 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
<i>Frames</i>	<b>256</b>	<b>512</b>
32	6.65 / 110%	5.75 / 95 %
64	5.17 / 86%	6.72 / 112%
128	6.64 / 107%	7.33 / 122%
Fractalpar2 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
<i>Frames</i>	<b>256</b>	<b>512</b>
32	5.60 / 70%	7.02 / 87 %
64	6.63 / 82%	8.20 / 102%
128	8.02 / 100%	7.40 / 92%

Error



Fractalpar2 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
<i>Frames</i>	<b>256</b>	<b>512</b>
32	2.75 / 45%	2.99 / 49%
64	2.50 / 41%	2.88 / 48%
128	3.0 / 51%	3.0 / 51%
Fractalpar2 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
<i>Frames</i>	<b>256</b>	<b>512</b>
32	2.90 / 36%	2.89 / 36%
64	2.91 / 36%	2.85 / 38%
128	3.22 / 40%	3.0 / 35%

Success



# Resultados Obtidos

Fractalpar1  
Fractalpar2

Fractalpar1 - Thread(s) 2		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	1.78 / 89%	1.71 / 85 %
64	1.93 / 96%	1.97 / 98%
128	1.81 / 90%	2.0 / 100 %

Fractalpar1 - Thread(s) 4		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	3.21 / 80%	3.02 / 75 %
64	3.05/ 76%	3.16 / 79%
128	3.21 / 80%	3.12 / 78 %

Fractalpar1 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	3.15 / 52%	3.11 / 51 %
64	3.17/ 52%	3.08 / 51%
128	3.21 / 53%	3.22 / 53 %

Fractalpar1 - Thread(s) 8		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	3.02/ 37%	3.05 / 38 %
64	3.10/ 38%	3.17 / 39%
128	3.23 / 40%	3.78 / 47.28%

Fractalpar2 - Thread(s) 2		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	1.96 / 98%	2.02 / 101 %
64	1.69 / 79%	1.96 / 98%
128	2.08 / 104%	2.03 / 101 %

Fractalpar2 - Thread(s) 4		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	2.41 / 60%	2.74 / 69 %
64	2.78 / 69 %	2.85 / 71%
128	3.16 / 79%	2.94 / 73%

Fractalpar2 - Thread(s) 6		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	2.75 / 45%	2.99 / 49%
64	2.50 / 41%	2.88 / 48%
128	3.0 / 51%	3.0 / 51%

Fractalpar2 - Thread(s) 8		
Speedup (~=2) / Eficiência (%)		
Frames	256	512
32	2.90 / 36%	2.89 / 36%
64	2.91 / 36%	2.85 / 38%
128	3.22 / 40%	3.0 / 35%



# Referências

- Shared and private variables in a parallel environment. <https://ibm.co/2VS9zTe>
- Conjunto de Mandelbrot. <https://bit.ly/2UD3DRC>
- OpenMP: Data-Sharing Rules. <https://bit.ly/2NJd6zX>
- OpenMP. <https://bit.ly/2PehiJ6>



# Programação Paralela

## Perguntas?

Adriano Luís de Almeida - [alalmeida@inf.ufsm.br](mailto:alalmeida@inf.ufsm.br)

**UFSM - Sistemas de Informação**  
**Abril de 2019**