

Meno: Peter Berta

Dátum: 29.11.2016

Predmet: Paralelné programovanie

Cvičenie: Streda 10:00

Cvičiaci: Matej Kloska

Projekt: Využitie OpenMP a MPI pri vykresľovaní fraktálu

## 1. Fraktál

Na vypočítanie bodov fraktálu som využil algoritmus zo stránky <http://jonisalonen.com/2013/lets-draw-the-mandelbrot-set/>. Jedná sa o Mandelbrotov fraktál. Farby sú upravené tak, aby bolo vidieť jednotlivé iterácie.

## 2. OpenGL

Ako základ pre riešenie tohto projektu som použil kód z programu „gl\_gradient.c“ zo vzorových príkladov poskytnutých na začiatku semestra.

## 3. MPI

Pomocou MPI rozosielať procesom inštrukcie ohľadom výpočtov bodov fraktálu. Jeden hlavný proces sa stará o rozosielanie týchto inštrukcií ostatným procesom a tiež o opätovné spojenie obrázku dokopy, teda prijatie vypočítaných dát.

```
err = MPI_Send(msgf, 6, MPI_FLOAT, i, tag, MPI_COMM_WORLD);
```

Medzi tieto inštrukcie patrí: začiatok výpočtu, koniec výpočtu, počet iterácií, priblíženie, posun na vodorovnej osi a posun na zvislej osi. Začiatok a koniec výpočtu predstavujú začiatok a koniec buffera, ktorý má daný proces vypočítať.

```
start = (int)msgf[0];  
end = (int)msgf[1];  
max = msgf[2];  
zoom = msgf[3];  
movex = msgf[4];  
movey = msgf[5];
```

Všetky procesy okrem hlavného sa vykonávajú v cykle, teda stále čakajú na nejakú správu. Toto čakanie je ukončené, keď dostanú správu, v ktorej sa prvá inštrukcia rovná -1.

## 4. OpenMP

Pomocou OpenMP rozdeľujem časť buffera, ktorú má daný proces vypočítať.

```
#pragma omp parallel for collapse (2) private(row, col) shared(image)
```

Príkazy „for collapse(2)“ oba optimalizujú funkcionality OpenMP pre paralelizáciu cyklov. Premenné „row“ a „col“ sú určené ako privátne, pretože každý proces ich má iné. Buffer „image“ je však spoločný a viaceré procesy doňho zapisujú a preto je označený ako zdieľaný.

```
.. num_threads(4)
```

Počet vlákien nastavíme pridaním príkazu „num\_threads(n)“ na koniec príkazu „#pragma...“.

## 5. Výsledky pozorovania

Pre účely testovania bola určená konštantná hĺbka fraktálu = 30000.

| Merania | 1 proces |          |          |          | 2 procesy |          |          |          |
|---------|----------|----------|----------|----------|-----------|----------|----------|----------|
|         | 1 thread | 2 thread | 4 thread | 8 thread | 1 thread  | 2 thread | 4 thread | 8 thread |
| 1.      | 17,428   | 9,487    | 9,881    | 8,265    | 8,914     | 8,758    | 7,602    | 6,600    |
| 2.      | 17,414   | 9,402    | 9,548    | 8,488    | 8,675     | 8,766    | 7,524    | 5,905    |
| 3.      | 17,402   | 9,370    | 9,718    | 8,492    | 9,700     | 8,819    | 7,541    | 5,918    |
| 4.      | 17,365   | 9,317    | 9,742    | 8,460    | 8,680     | 8,860    | 7,550    | 5,930    |
| 5.      | 17,369   | 9,145    | 9,745    | 8,373    | 8,996     | 8,770    | 7,580    | 5,960    |
|         | 17,396   | 9,344    | 9,727    | 8,416    | 8,993     | 8,795    | 7,559    | 6,063    |

| Merania | 4 procesy |          |          |          | 8 procesov |          |          |          |
|---------|-----------|----------|----------|----------|------------|----------|----------|----------|
|         | 1 thread  | 2 thread | 4 thread | 8 thread | 1 thread   | 2 thread | 4 thread | 8 thread |
| 1.      | 10,215    | 8,533    | 6,539    | 6,174    | 8,330      | 6,193    | 5,734    | 5,678    |
| 2.      | 10,328    | 8,437    | 6,242    | 6,117    | 8,160      | 6,770    | 6,631    | 5,419    |
| 3.      | 10,108    | 8,428    | 6,125    | 6,112    | 8,156      | 6,770    | 5,856    | 5,390    |
| 4.      | 10,227    | 8,842    | 6,358    | 6,100    | 8,128      | 6,490    | 5,754    | 5,428    |
| 5.      | 10,119    | 8,494    | 6,329    | 6,128    | 8,645      | 6,500    | 5,663    | 5,552    |
|         | 10,1994   | 8,5468   | 6,3186   | 6,1262   | 8,2838     | 6,5446   | 5,9276   | 5,4934   |



