

Dokumentácia – paralelizované generovanie fraktálov

Popis riešenia – Generovanie fraktálov bolo paralelizované za použitia OpenMP v jazyku C++. V programe sú použité 2 paralelné oblasti, jedna na vytvorenie teamu threadov (čerpala som z cvičení) a druhá pre jednotlivé thready pri funkcionalite vykreslovania. Vykreslovanie fraktálu beží pod touto funkcionalitou mimo paralelných oblastí.

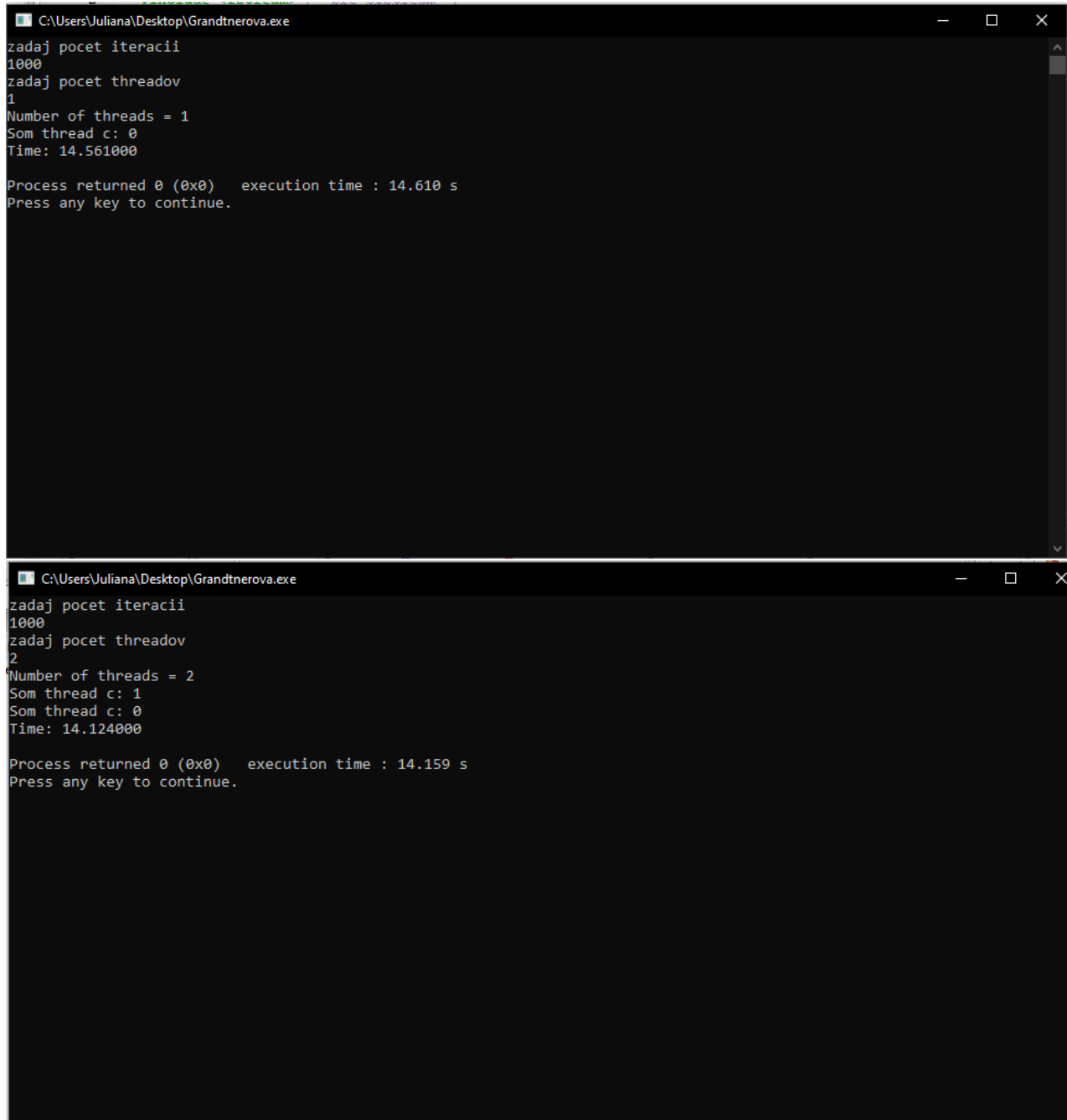
Kostru funkcionality programu som nevytvorila, našla som ju dostupnú vo viacerých internetových zdrojoch a spočíva v obsahu druhej paralelnej oblasti a classoch:

- class linear2d_function
- class pixel

Program obsahuje aj možnosť vybrať si počet threadov, s ktorým chcem, aby procesor pri vytváraní programu pracoval, a aj počet iterácií, vykonávaných pri generovaní pixela.

Príklad výsledkov paralelizácie

Počet iterácií je pri jednom kole 1000, pri druhom kole 100 (pre porovnanie), počet threadov 1,2,4,8

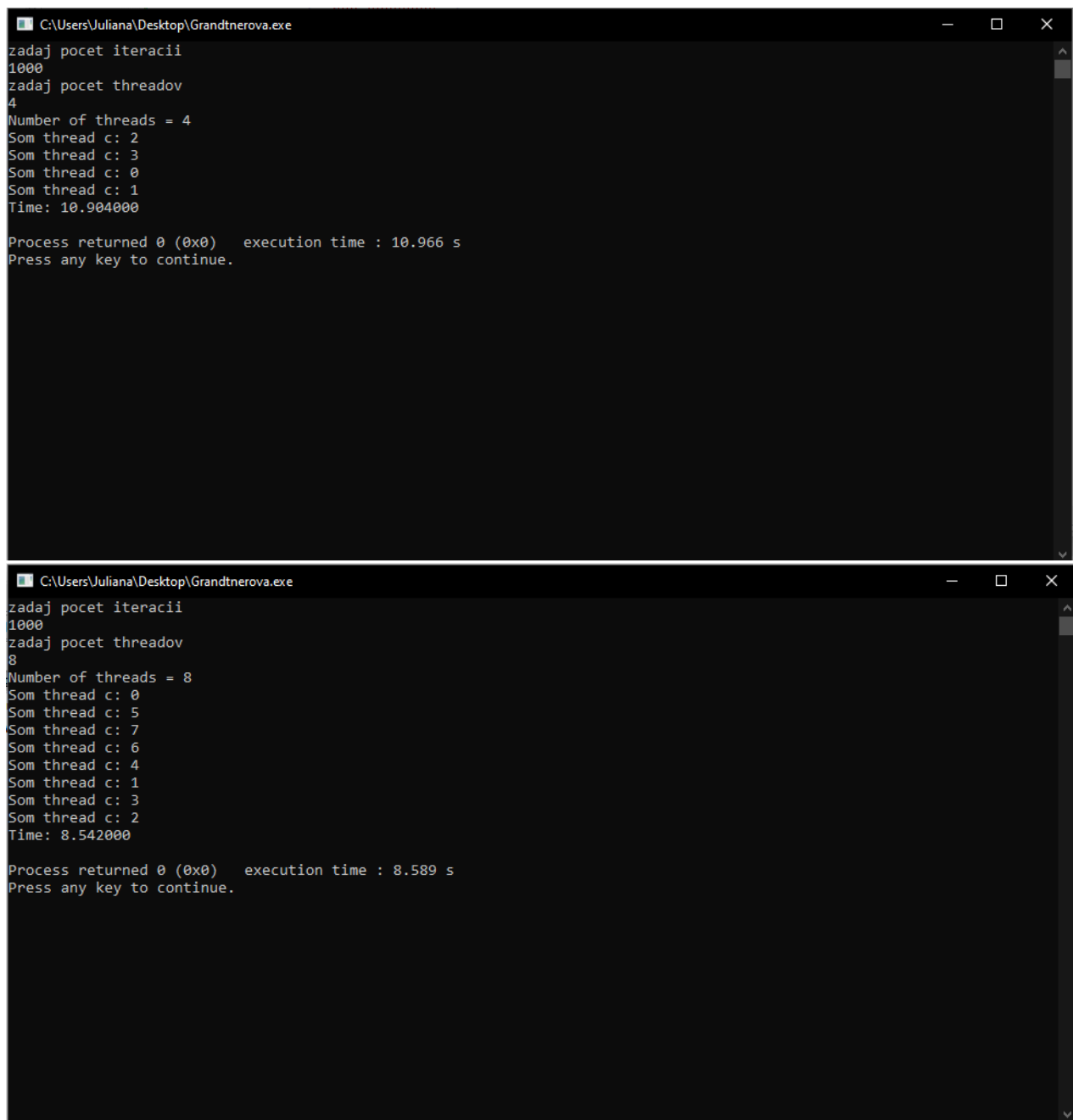


```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
1000
zadaj pocet threadov
1
Number of threads = 1
Som thread c: 0
Time: 14.561000

Process returned 0 (0x0)   execution time : 14.610 s
Press any key to continue.

C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
1000
zadaj pocet threadov
2
Number of threads = 2
Som thread c: 1
Som thread c: 0
Time: 14.124000

Process returned 0 (0x0)   execution time : 14.159 s
Press any key to continue.
```



```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
1000
zadaj pocet threadov
4
Number of threads = 4
Som thread c: 2
Som thread c: 3
Som thread c: 0
Som thread c: 1
Time: 10.904000

Process returned 0 (0x0)   execution time : 10.966 s
Press any key to continue.

C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
1000
zadaj pocet threadov
8
Number of threads = 8
Som thread c: 0
Som thread c: 5
Som thread c: 7
Som thread c: 6
Som thread c: 4
Som thread c: 1
Som thread c: 3
Som thread c: 2
Time: 8.542000

Process returned 0 (0x0)   execution time : 8.589 s
Press any key to continue.
```

Prvé kolo testovania ukázalo, že rýchlosť sa zvyšovaním vlákien postupne znižovala (v ďalších 2 testoch boli výsledky pre jednotlivé thready podobné, avšak jedenkrát sa stalo, že pri 4 threadoch a 1000 iteraciách mi zamrzol codeblocks – následne som testy pre 4 thready robila ešte párkrát a už sa to nestalo).

Druhé kolo testovania (obrázky nižšie) však ukázalo, že pri 4 a 8 threadoch pri 100 iteráciách (odskúšané viackrát) hodnota zostávala vždy vyššia ako pri 2 threadoch – 2 thready bol ideál.

Neviem však úplne presne, prečo to tak vyšlo.

```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
100
zadaj pocet threadov
1
Number of threads = 1
Som thread c: 0
Time: 4.515000

Process returned 0 (0x0)   execution time : 4.547 s
Press any key to continue.
```

```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
100
zadaj pocet threadov
2
Number of threads = 2
Som thread c: 1
Som thread c: 0
Time: 3.434000

Process returned 0 (0x0)   execution time : 3.481 s
Press any key to continue.
```

```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
100
zadaj pocet threadov
4
Number of threads = 4
Som thread c: 2
Som thread c: 3
Som thread c: 0
Som thread c: 1
Time: 3.828000

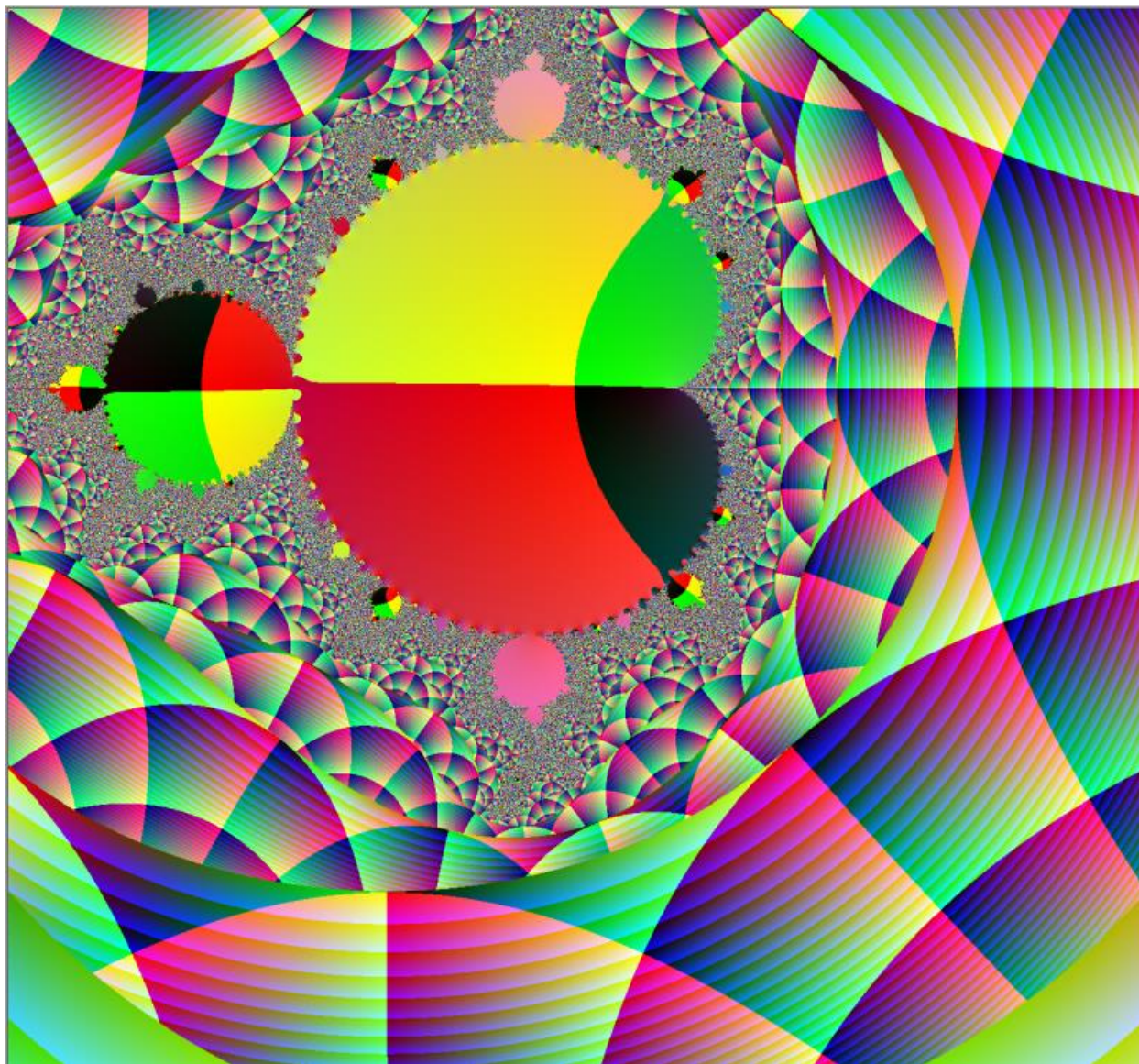
Process returned 0 (0x0)   execution time : 3.859 s
Press any key to continue.
```

```
C:\Users\Juliana\Desktop\Grandtnerova.exe
zadaj pocet iteracii
100
zadaj pocet threadov
8
Number of threads = 8
Som thread c: 7
Som thread c: 4
Som thread c: 5
Som thread c: 6
Som thread c: 3
Som thread c: 1
Som thread c: 0
Som thread c: 2
Time: 3.555000

Process returned 0 (0x0)   execution time : 3.595 s
Press any key to continue.
```

Výstupy pre 100 a 1000 iterácií sú mierne rôzne:

100:



1000:

