CUDA exercise 2: Mandelbrot Set

1 Timers

Timers can be used to compare the performance of the CPU and the GPU.

1.1 CUDA Syntax

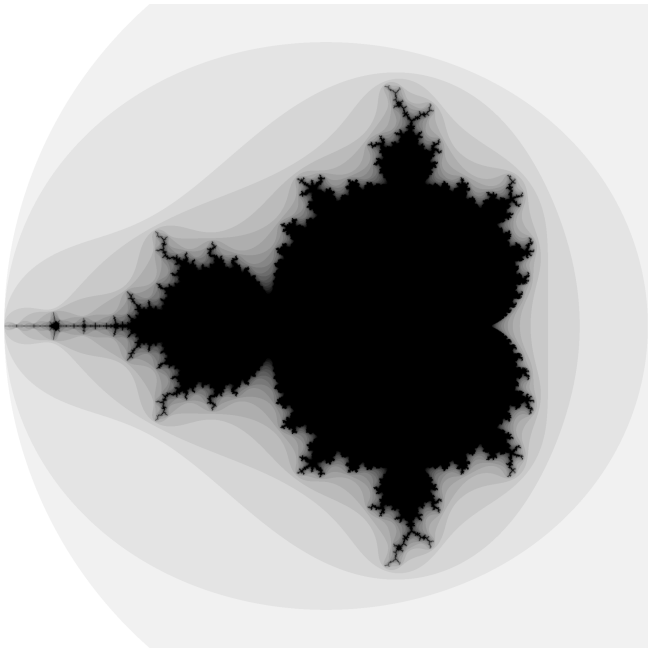
```
StopWatchInterface* timer_gpu = NULL;
sdkCreateTimer(&timer);

sdkStartTimer(&timer);
....
sdkStopTimer(&timer);

float time = sdkGetAverageTimerValue(&timer); // average elapsed time in ms
float time = sdkGetTimerValue(&timer); // overall elapsed time in ms

sdkResetTimer(&timer);
sdkDeleteTimer(&timer);
```

2 Mandelbrot set



Consider the complex sequence defined by $z_{n+1} = z_n^2 + c$ and $z_0 = 0$ with $c \in \mathbb{C}$, $z \in \mathbb{C}$ and $n \in \mathbb{N}$. The Mandelbrot set is the set of numbers c such that the sequence z_n remains bounded for all $n \in \mathbb{N}$. It can be shown that the sequence diverges

if for any n $|z_n| > 2$. This property is useful for representing an approximation of the Mandelbrot set: the number n of iterations until $|z_n| > 2$ for a grid of values c is used to color the complex plane.

2.1 Kernel

Write the kernel `mandelbrotKernel` similar to the computation on the host and call it in `main`. Verify that the results for `niter=20` are correct.
Run the program with `niter=40`. Are the GPU results correct? Explain the results.

2.2 Timers

Add 2 timers `timer_gpu` and `timer_mt` to measure the time needed to do the calculation on the GPU and to copy the results from the device to the host. Calculate the speedup with and without the memory transfer.
How does reducing or increasing `niter` affect the speed up? Explain.
Modify the number of threads per block and the number of blocks in the grid. What happens?