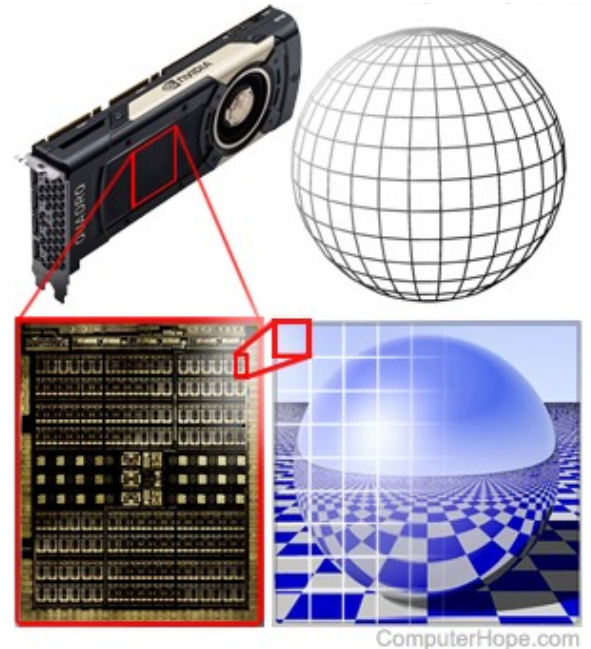


Embarrassingly parallel

Updated: 06/30/2019 by Computer Hope

A computing task is **embarrassingly parallel** (also called **pleasingly parallel** or **perfectly parallel**) if it can naturally divide into individual parallel processing tasks. Many image processing tasks, such as raytracing and raycasting of 3D images, are embarrassingly parallel. In these tasks, sections of the image can be computed separately, without interacting with the others. These individual "chunks" of the image can be assigned to independent processors, such as the CUDA cores on an NVIDIA GPU. Each core can process its chunk simultaneously, greatly speeding up the process. When parallel processing is complete, the chunks are joined to create the final rendered image.



Pictured at right is an illustration of an NVIDIA GPU equipped with multiple CUDA cores, each processing part of a raytraced image in parallel.

Examples of embarrassingly parallel tasks

In addition to 3D image rendering, other embarrassingly parallel tasks include:

- **Genetic programming** — In genetic programming, algorithms are "evolved" in groups by combining and mutating previous generations. Each of these individuals are processed independently in parallel.
- **Mandelbrot set** — A fractal where each point is calculated individually, independent of the others.
- **Monte Carlo algorithms** — A wide range of computational tasks that are processed by pseudorandom sampling of individual elements.
- **DFT (Discrete Fourier Transform)** — A widely-used technique for processing equally-spaced samples of a function or signal. Commonly used in DSP (digital signal processing), based on the

work of French mathematician Jean Fourier.

- **Brute-force cryptographic tasks** — A range of cryptographic problems, such as Proof-of-Work algorithms, that perform many similar computations, efficiently executed in parallel.