> Most cores have two threads

Here, I think you are confusing the overloaded term "thread". As correctly pointed out by other answers, a thread usually refers to a "software" concept. But sometimes it is also used as a "hardware" concept. When a "core" has two "threads" (like in many new Intel chips), it means that the core can run two parallel threads, as if there were two cores. This is, however, usually called hyperthreading. See:

http://en.wikipedia.org/wiki/Hyper-threading

So if you have N threads (I mean software threads, those made within your application, or just by running different applications at the same time) and M processors (being cores, or the hardware threads explained above), the following happens:

- N<=M: then the Operating System should assign each thread a different processor. Then the application threads run truly in parallel.
- N>M: then some threads have to share a processor.