

第十二次作业

1. 并行的加法

问题描述

编写一个程序，采用多线程的方法计算一个数组的各个元素之和。具体来说，该数组为整型数组 `a`，有 300 个元素。在 `main` 函数中，对该数组的每个元素进行赋值，值为一个 1-100 的随机数。然后创建 3 个线程，第 1 个线程负责计算 `a[0]~a[99]` 这 100 个元素之和；第 2 个线程负责计算 `a[100]~a[199]` 这 100 个元素之和；第 3 个线程负责计算 `a[200]~a[299]` 这 100 个元素之和。最后在 `main` 函数中汇集各个线程的计算结果，然后算出最终的结果，即 `a[0]~a[299]` 这 300 个元素的总和，并且打印出来。

注：

1. 主类名为 `ParallelSum`，包名为 `parallelsun`。
2. 需要解决多个线程之间的数据共享及同步问题，不能仅使用 `sleep()` 函数等待线程执行结束。

2. MatrixCalculator

随机生成两个 `1000*1000` 的 `double` 型的矩阵 `M1` 和 `M2`，每个元素的大小小于等于 100。用串行算法计算 `M1*M2` 的结果，并且输出程序串行计算的时间。再开 4 个线程并行计算 `M1*M2` 的结果，并且输出程序并行计算的时间。比较这两次计算的结果，如果计算结果相同输出 “YES”，否则输出 “NO”。

提示：四个线程可以分别计算 0-249, 250-499, 500-749, 750-999 行的矩阵值。

3. 疯狂的打印机

问题描述

将第十一次作业中“疯狂的打印机”的两个线程设置优先级，先启动的线程使用默认优先级，后启动的线程设置优先级为 8，再执行一次，观看显示结果。

注：将本次控制台输出的结果保存在 `writer8.txt` 中，并与之前的 `writer.txt` 进行比较。