并行计算

第十一次课作业

姓名 陆子毅 学号 3022206045

请简要回答以下问题：

1、计算密集型算法与数据密集型算法对并行计算环境有什么不同的要求？

计算密集型算法：这类算法的特点是执行大量的计算操作，对处理器的计算能力要求较高。在并行计算环境中，计算密集型算法需要强大的计算能力，例如多核处理器、GPU或其他高性能计算设备，以便同时执行大量的计算任务。此外，它们可能还需要有效的任务调度和负载均衡机制，以确保所有计算资源得到充分利用。

数据密集型算法：这类算法处理的数据量巨大，对内存和存储系统的要求较高。在并行计算环境中，数据密集型算法需要高速的数据传输能力和大容量的内存或存储空间，以便快速读写大量数据。此外，它们可能还需要优化的数据分布和访问策略，以减少数据传输的延迟和带宽消耗。

2、PPT第36页的并行求前缀和算法，与第11页的数组求和算法有什么关系？

它们都是处理数组数据的并行算法，但目标和方法有所不同。

数组求和算法：目标是计算整个数组所有元素的总和。在并行化的过程中，可以通过将数组分割成多个子数组，并在多个处理器上独立计算每个子数组的和，最后再将这些局部和合并起来得到最终结果。

并行求前缀和算法：这个算法的目标是计算数组中每个元素的前缀和，即从数组的第一个元素到当前元素的累积和。并行求前缀和算法通常涉及到多个处理器之间的通信和数据交换，因为每个处理器需要获取相邻处理器计算的前缀和结果，以便正确计算自己的累积和。

3、并行程序设计方法都有哪些，请举例予以说明。

（1）直接并行化串行算法：分析现有的串行算法，发掘和利用其中的并行性，直接将其改造为并行算法。例如，课程中提到的数组求和算法，可以通过数据分解、按块均分和分治的方法，并行化执行。

（2）从问题描述开始设计并行算法：根据问题本身的描述和固有属性，从头设计一个全新的并行算法。课程中的求前缀和算法就是一个例子，它通过并行计算每个元素的累积和，并在处理器之间进行有效的通信和数据交换来实现。

（3）借用已有算法求解新问题：找出新问题和某个已解决问题之间的联系，改造或利用已知的并行算法来解决新问题。课程中提到的3PCF问题（三点相关函数），通过矩阵乘法和比较操作来解决，就是一个利用已有算法的例子。