#### 分值： 填空题（16‘）+ 简答题（4 \* 6’）+ 计算题（3 \* 10‘）+ 程序设计题（3 \* 10’）

#### 填空题

#### 简答题

##### 并行计算结构模型：

（1）结构类型

* SISD：单指令流单数据流计算机（冯诺依曼机） Single-Instruction Multiple-Data
* SIMD：单指令流多数据流计算机
* MISD：多指令流单数据流计算机
* MIMD：多指令流多数据流计算机

（2）几种MIMD

* PVP：并行向量处理机（Parallel Vector Processor）：多VP（向量处理器）通过交叉开关和多个SM（共享内存）相连
* SMP：对称多处理机（Symmetric Multiprocessor）：多P/C（商品微处理器）通过交叉开关/总线和多个SM（共享内存）相连
* MPP：大规模并行处理机（Massively Parallel Processor）：处理节点有商品微处理器+LM（分布式本地内存），节点间通过高带宽低延 迟定制网络互联，异步MIMD，多个进程有自己的地址空间，通过消息传递机制通信
* COW：工作站机群（Cluster Of Workstations）：节点是完整操作系统的工作站，且有磁盘
* DSM：分布共享存储处理机（Distributed Shared Memory）：高速缓存目录DIR确保缓存一致性，将物理分布式LM组成逻辑共享SM从 而提供统一地址的编程空间

注：对称指所有处理器都能同等地访问I/O很同样的运行程序（如OS和I/O服务程序），而非对称主从式是 仅有主处理器运行OS和控制访问I/O并监控从处理器执行

##### 访问存储模型

* UMA（Uniform Memory Access）均匀存储访问：物理存储器被所有处理器均匀共享，所有处理器对所有SM访存时间相同，每台处理器可带有高速私有缓存，外围设备共享。
* NUMA非均匀存储访问：共享的SM是由物理分布式的LM逻辑构成，处理器访存时间不一样，访问LM或CSM（群内共享存储器）内存储器比访问GSM（群间共享存储器）快
* COMA（Cache-Only MA）全高速缓存存储访问：NUMA的特例、全高速缓存实现
* CC-NUMA（Coherent-Cache NUMA）高速缓存一致性NUMA：NUMA＋高速缓存一致性协议
* NORMA（No-Remote MA）非远程存储访问：无SM，所有LM私有，通过消息传递通信

##### 系统互联

总线、SAN（System Area Network，系统域网络）、LAN（Local Area Network，局域网）、MAN（Metropolitan Area Network，都域网）、WAN（Wide Area Network，广域网）

处理器总线、局部总线、存储器总线；I/O总线、系统总线

##### 网络性能指标

* 节点度（Node Degree）：射入或射出一个节点的边数。在单向网络中，入射和出射边之和称为节点度。
* 网络直径(Network Diameter)：网络中任何两个节点之间的最长距离，即最大路径数。
* 对剖宽度（Bisection Width）：对分网络各半所必须移去的最少边数。
* 对剖带宽（Bisection Bandwidth）：每秒钟内，在最小的对剖平面上通过所有连线的最大信息位（或字节）数。
* 如果从任一节点观看网络都一样，则称网络为对称的(Symmetry)。

##### 静态互连

处理单元间有固定连接的网络，程序执行期间这种点到点的连接不变

* 一维线性阵列LA/LC：二邻近串联
* 二维网孔MC：四邻近连接（Illiac连接、2D环绕）
* 树连接TC：二叉树、星型网络、二叉胖树（节点通路向根节点方向逐渐变宽，解决通信瓶颈）
* 超立方HC：3立方、4立方
* 立方环：3立方顶点用环代替

静态互连网络特性比较：

##### 动态互连

交换开关构成的，可按应用程序要求动态改变连接组态

* 总线：连接处理器、存储模块、I/O外围设备等的一组导线和插座，分时工作、多请求总线仲裁，多总线（本地、存储、数据、系统）和多层总线（板级、底板级、I/O级）
* 交叉开关：高带宽的开关控制的专用连接通路网络，NxN的开关网络同时只能接通N对源目的通信
* 多级互联网络MIN：每一级用多个开关单元，各级之间有固定的级联拓扑

##### 标准网络互联

* FDDI光纤分布式数据接口
* 快速以太网
* Myrinet：商用千兆位包开关网
* InfiniBand：交换式通信结构

##### 性能指标

##### 并行算法的复杂度

* 运行时间t(n)：求解问题的时间，包括计算时间和通信时间
* 处理器数目p(n)：求解给定问题所用的处理器数目
* 成本c(n)：c(n) = t(n)\*p(n)
* 成本最优——并行算法的成本在数量级上等于最坏情况下串行求解次问题所需要的执行步数
* 工作量W(n)：并行算法完成的总的操作数量
* 工作量最优：功耗低、环保

并行算法的WT表示——Brent定理：令W(n)是并行算法A在运行时间T(n)内所执行的运算量，则A使用p台处理器可在t(n) = O(W(n)/p + T(n))时间内执行完毕

##### 并行算法设计过程 PCAM：划分，通信，组合，映射

例子：求前缀和

PCAM步骤：划分→通讯→组合→映射

划分：分解成小的任务，开拓并发性。

* 先进行**域分解**，再进行**功能分解**
* 域分解：划分的对象为**数据**。可以是算法的输入数据、中间处理数据和输出数据。将数据分解成大致相等的小数据片。如果一个任务需要别的任务中的数据，则会产生任务间的通讯。
* 功能分解：划分的对象为**计算**。划分后，如果不同任务所需数据不想交则划分成功；如果数据有相当的重叠，则需要重新进行域分解和功能分解。

通讯：确定诸任务间的数据交换，检测划分的合理性。

* 划分产生的诸任务，一般不能完全独立执行，需要在任务间进行数据交流，从而产生了通讯。
* 功能分解确定了诸任务之间的数据流
* 诸任务是并发执行的，通讯限制了这种并发性
* 四种通讯模式  
   局部/全局通讯：局部通讯限制在一个邻域内，全局通讯则是非局部的。  
   结构化/非结构化通讯：结构化通讯下每个任务的通讯模式是相同的  
   静态/动态通讯  
   同步/异步通讯

组合：依据任务的局部性，组合成更大的任务。

* 合并小尺寸任务，减少任务数。如果任务数恰好等于处理器数，则也完成了映射过程。
* 通过增加任务的粒度和重复计算，可以减少通讯成本。
* 保持映射和扩展的灵活性，降低软件工程成本。  
  表面-容积效应：通讯量和任务子集的表面成正比，计算量和任务子集的体积成正比。增加重复计算有可能减少通讯量→重复计算减少了通讯量，但增加了计算量，应保持恰当的平衡，重复计算的目标应当是减少算法的总运算时间。(实例：二叉树求和，碟式结构使用了重复计算，但减少了总时间)

映射：将每个任务分配到处理器上，提高算法的性能。

* 每个任务要映射到具体的处理器，定位到运行机器上。
* 任务数大于处理器数时，存在负载均衡和任务调度问题。
* 映射的目标：减少算法的总执行时间
* 基本原则：并发的任务映射到不同的处理器上，存在高通讯的任务则尽量映射到相同的处理器上。
* NPC

##### 并行计算模型

**1.PRAM模型（SIMD-SM）**  
PRAM（Parallel Random Access Machine）并行随机存取机器，是一种抽象并行计算模型，它假设：

* 存在容量无限大的SM
* 有限或无限个功能相同的处理器，且均有简单算术运算和逻辑判断功能
* 任何时刻各处理器可通过SM交换数据

根据并发访问机制，又分为：

* 不允许同时读和同时写的PRAM-EREW
* 允许同时读但不允许同时写的PRAM-CREW
* 允许同时读和同时写的PRAM-CRCW

PRAM-CRCW又分为：

* 只允许同时写相同的数CPRAM-CRCW
* 只允许优先处理器先写PPRAM-CRCW
* 允许任意处理器自由写APRAM-CRCW

PRAM优点：

* 适合并行算法表达、分析和比较
* 使用简单，屏蔽了通信、存储管理、进程同步等并行细节
* 易于修改算法设计以适应不同并行机

PRAM缺点：

* PRAM是同步模型，同步锁费时
* 不适用于MIMD和DM
* 假设任何处理器可在单位时间内访问任何处理单元而不考虑竞争和带宽，不现实

**2.异步APRAM模型（MIMD-SM）**  
异步APRAM模型假设：

* 每个处理器有LM、局部时钟、局部程序
* 处理器通信经过SM
* 无全局时钟，各处理器异步执行各自指令
* 处理器之间的指令相互依赖关系必须显式加入同步障
* 一条指令可以在非确定但有界时间内完成

指令类型有：

* 全局读：从SM读到LM
* 局部操作：LM操作存入LM
* 全局写：LM写入SM
* 同步：各处理器需等到其他处理器到达后才能继续执行

APRAM比PRAM更加接近实际并行机

**3.BSP模型（MIMD-DM）**  
BSP（Bulk Synchronous Parallel）大同步并行机（APRAM算作轻量）是一个分布式存储的MIMD模型，它的计算由若干全局同步分开的、周期为L的超级步组成，各超级步中处理器做LM操作并通过选路器接收和发送消息；然后做一次全局检查，以确定该超级步是否已经完成（块内异步并行，块间显式同步）  
参数：处理器数p、选路器吞吐率g、全局同步间隔L、一个超级步中一个处理器至多发送或接收h条消息

**4.LogP模型：MIMD-DM，点到点通讯**  
LogP模型是分布式存储、点到点通信的MIMD模型  
LogP采取隐式同步，而不显式同步障

##### 可扩放性度量标准（1）

* 可扩放性最简朴的含意是在确定的应用背景下，计算机系统（或算法或程序等）性能随处理器数的增加而按比例提高的能力
* 影响加速的因素有：1. 求解问题中的串行分量，2. 并行处理所引起的額外开销（通信、等待、竞争、冗余操作和同步等），3．加大的处理器数超过了算法中的并发程度。
* 增加问题的规模有利于提高加速的因素是：1. 较大的问题规模可提供较高的并发度，2. 额外开销的增加可能慢于有效计算的增加，3. 算法中的串行分量比例不是固定不变的（串行部分所占的比例随着问题規模的增大而缩小).
* 增加处理器数是会增加额外开销和降低处理器的利用率的，所以对于一个特定的并行系统、并行算法或并行程序，它们能否有效利用不断增加的处理器的能力应是受限的，而度量这种能力就是可扩放性这一指标。

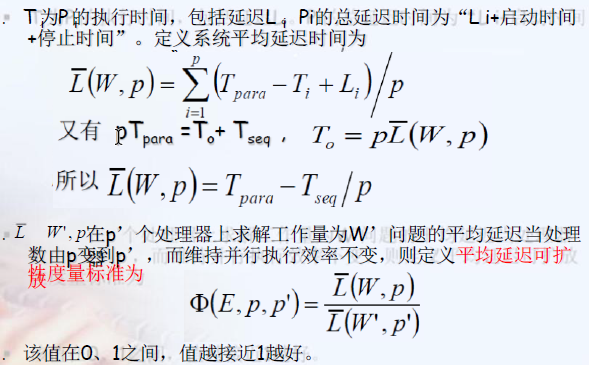
##### 可扩放性度量标准（2）

* 可扩放性:调整什么和按什么比例调整  
  . 并行计算要调整的是处理数p和问题规模W，  
  . 两者可按不同比例进行调整，此比例关系（可能是线性的，多项式的或指数的等）就反映了可扩放的程度。
* 与并行算法和体系结构相关
* 可扩放性研究的主要目的：
* . 确定解决某类问题用何种并行算法与何种并行体系结构的组合，可以有效地利用大量的处理器；  
  . 对于运行于某种体系结构的并行机上的某种算法当移植到大规模处理机上后运行的性能；  
  . 对固定的问题规模，确定在某类并行机上最优的处理器数与可获得的最大的加速比；  
  . 用于指导改进并行算法和并行机体系结构，以使并行算法尽可能地充分利用可扩充的大量处理器；
* 目前无一个公认的、标准的和被普遍接受的严格定义和评判它的标准

##### 等效率度量标准

##### 等速度度量标准

##### 平均延迟度量标准



##### 基本设计技术

* 划分：均匀划分PSRS
* 分治：双调归并网络
* 流水线：五点DFT计算、4流水线编程

##### OpenMP MPI CUDA 特点 使用场合

####

#### 计算题

#### 程序设计题

‌填空题 1 16  
‌简答题 4 6  
基础盖帘，分类6类  
存储访问类型，ccnuma。。。  
处理机结构，线性，网格，二维，3维。对剖宽度  
加速比，阿姆达尔，限制  
并行算法分析指标  
并行算法设计过程，通信，组合，映射  
并行算法计算模型，bsp，异步buram  
可阔放心度量标准，3个，特点  
基本设计技术，第七章，划分，分之，流水线  
opmt，mpi，cuda，特点，使用场合

‌

计算题 10 3  
课后作业，书4.6 4.7 ，cpi，mps，ppt  
4.11加速比计算，阿姆达尔，串行风量，机器需求，极限，10台机器能加速到多少  
5.2算复杂度，类似，界  
brant定理证明  
第七章 并行划分，psrs算法，数据根据算法写步骤结果  
openmt

‌程序设计 10 3  
mpi 补充，填，最基本函数6个，p411页，p412  
cuda 6行 求3个数加法，填空，分配内存，内存拷贝，核函数线程编号索引，block，函数  
openmp 并行化，联系过的题，计算圆周率转化为并行。398 399 401