《并行分布计算》复习提纲 （2021.12）**题型：单选、填空、判断、分析、论述、综合√**

大题就两种题型：1.比较有什么区别 2.发展过程是什么 包括哪些内容 相互之间有什么关系

1. 并行计算机互联网络

静态互连网络：

处理单元间有着固定连接的一类网络，在程序执行期间，这种点到点的链接保持不变；典型的静态网络有一维线性阵列、二维网孔、树连接、超立方网络、立方环、洗牌交换网、蝶形网络等。

备注：

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

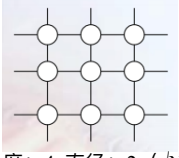
（1）一维线性网络：

每个节点只与左右相连，当首尾相连时构成循环位移器。度恒为2，直径恒为⌊N/2⌋（双向环）或N-1(单向环）

（2）二维孔网

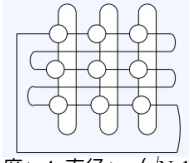
√N×√N二维网孔

（a）2-D网孔 节点与上下左右相连



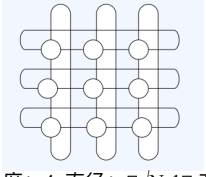
度：4 直径：2（√N-1） 对剖宽度为√N

（b）Illiac网孔 垂直方向上带环绕，水平方向呈蛇状首尾相连



度：4 直径：（√N-1） 对剖宽度为2√N

（c）2-D环绕 垂直和水平方向均带环绕



度：4 直径：⌊√N-1⌋ 对剖宽度为2√N

（d）二叉树（满）

度：3 对剖宽度：1 直径：$2(\lceil \log N \rceil - 1)$

（e） 超立方

N-超立方由$N=2^n$个顶点组成

度为n，直径n，对剖宽度n/2

动态互连网络：用交换开关构成的，可按应用程序的要求动态地改变连接组态；

典型的动态网络包括总线、交叉开关和多级互连网络等。

交叉开关：一个开关有n个输入和n个输出，只允许一对一或一对多映射

层级互联：

动态互连网络的复杂度和带宽性能一览表			
网络特性	总线系统	多级互连网络	交叉开关
硬件复杂度	$O(n+w)$	$O((n\log_k n)w)$	$O(n^2w)$
每个处理器带宽	$O(wf/n) \sim O(wf)$	$O(wf)$	$O(wf)$
最小时延	恒定(轻负荷)	$O((n\log_k n))$	恒定

标准互连网络：

光纤分布式数据接口

快速以太网

Myrinet

可扩放一致性接口(SCI)

InfiniBand

简答题1 冉令强例子必考

并行计算机结构模型经历了哪些阶段相互之间的关系

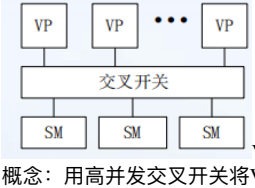
冉令强：5+1种模型 每种模型有什么特点 相互之间有什么界限关系

如DSM和MPP有什么关系 每种模型的关系弄清楚

几种模型的发展规律（按时间顺序发展 为什么会有这种发展顺序 原因是什么）

1.PVP（并行向量处理机）

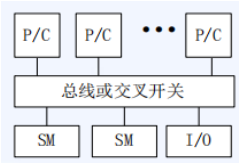
结构图

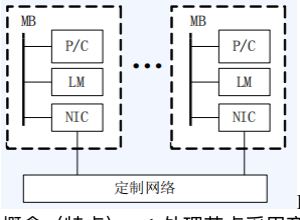


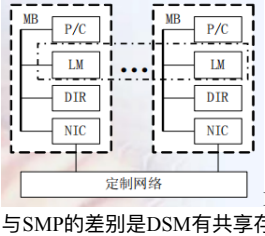
VP：向量存储器 SM：共享存储器

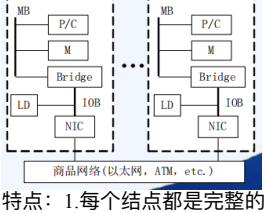
概念：用高并发交叉开关将VP连向共享存储

特点：不使用高速缓存，而是使用大量的向量寄存器和指令缓冲

优点：编程方便，效率高，早期无高速缓存时使用
2.SMP（对称多处理机器）
结构图


概念：使用商品微处理器，经由告诉总线连向SM，主要用于商用
优点：1.对称性2.单地址空间，易于编程3.支持数据的局部性4.低通信延迟
缺点：1.欠可靠2.客观的通信延迟：竞争会加剧延迟3.慢速增加的宽带4.不可括放性
3.MPP（大规模并行处理器）
结构图


P/C:处理器 LM:局部存储 NIC:网络接口芯片
概念（特点）：1.处理节点采用商品化的微处理器2.系统中有物理上的分布式处理器3.采用高速通信宽带和低延时的互联网络4.能扩散至成百上万个处理器5.是异步的MIMD机器
应用于科学计算等以计算为主的领域
缺点：只能同时访问一个节点的数据
4.DSM（分布共享存储处理机）
结构图


DIR:高速缓存目录
与SMP的差别是DSM有共享存储器，比MMP的优越性是编程更容易
缺点是节点无法做到太多
5.工作站机群（COW）
结构图


特点：1.每个结点都是完整的冯·诺依曼机2.各结点通过低成本的商品网络互连3.各结点内有本地磁盘，MMP没有4.节点内网络松散耦合到I/O总线5.完整的OS驻留在结点，基于UNIX

并行计算机访存模型

均匀存储访问(UMA)

特点

物理存储器被所有CPU均匀共享
所有CPU访问任何存储单元时间相同
每台CPU可带私有高速缓存
外围设备也可以以一定形式共享

紧耦合系统

适用于通用或分时应用
只有SM无LM

非均匀存储访问(NUMA)

特点

被共享存储器在物理上是分布在所有CPU中的，而所有LM的集合组成全局地址空间
CPU访问存储器时间是不一样的，访问LM或通过群内互连网络(CIN)访问群内共享存储器(CSM)较快；而访问外地存储器或通过全局互连网络访问全局共享存储器(GSM)较慢
每台CPU可带私有高速缓存，外设也可以通过某种形式共享

全高速缓存存储访问(COMA)

是NUMA一种特例
只是用Cache并使用其进行编址
没有存储层次结构
利用分布的高速缓存目录D进行远程高速缓存的访问

高速缓存一致性非均匀存储访问(CC-NUMA)

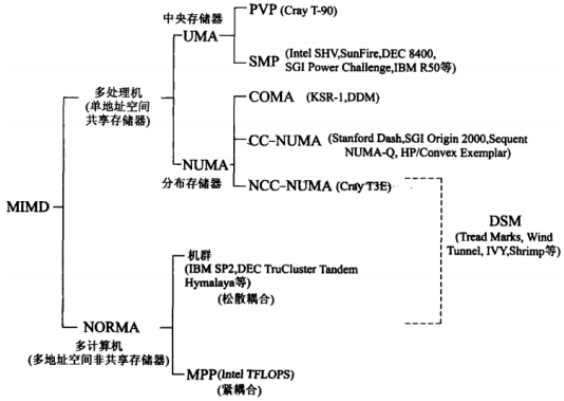
将一些SMP机器作为一个单节点而彼此连接起来所形成的一个较大的系统

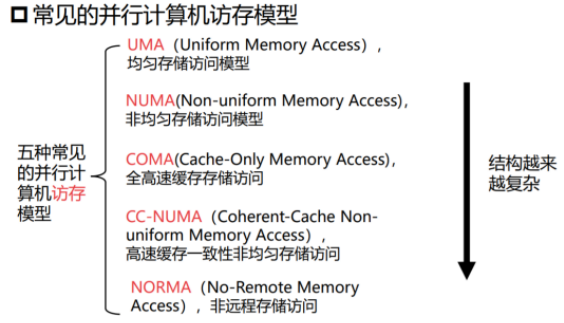
特点

使用基于目录的高速缓存一致性协议
改善常规SMP的可扩放性问题
实际上是一个分布共享存储的DSM多处理机系统

非远程存储访问(NORMA)

非远程存储访问模型
所有存储器是私有的，仅能由其所属的处理器访问
集群
绝大数NORMA都不支持远程存储器的访问
各处理器之间通过消息传递方式通信





模型的发展历程，关系，联系（13页）

大型并行机系统一般可分为6类机器：单指令多数据流计算机，并行向量处理机，对称多处理机，大规模并行处理机，工作站机群和分布共享存储多处理机。六种结构模型发展脉络SIMD计算机多为专用,其余的五种均属于多指令多数据流计算机。目前绝大多数近代并行机均用商品硬件构成，而PVP计算机的部件很多都是定制的。。

3.并行计算模型

150页，简答题2：四种模型的特点，区别 流程

特点：计算模型实就是硬件和软件之间的一种桥梁，使用它能够设计、分析算法,在其上高级语言能被有效地编译且能够用硬件来实现。

区别：抽象(如PRAM),要么过于专用(如互连网络模型和VLSI计算模型)。

（1）PRAM模型基本概念：（同6）

有一个集中的共享存储器和一个指令控制器，通过SM的R/W交换数据，隐式同步计算。又称SIMD-SM模型。

分类：

PRAM-EREW：互斥读、互斥写

PRAM-CREW 并发读、互斥写

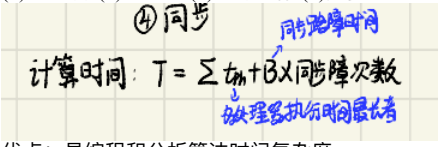
PRAM-CRCW并发读并发写：

CPRAM-CRCW：仅允许写入相同数据
PPRAM-CRCW：仅允许优先权较高的CPU写入
APRAM-CRCW：允许任意CPU自由写入
PRAM-CRCW是最强的计算模型，PRAM-EREW可倍模拟PRAM-CREW和PRAM-CRCW
优点：特别适合并行算法表达、分析和比较
使用简单，并行机的低级细节都隐含于模型
缺点：不适合分布存储的一部MIMD机器
(忽略SM的竞争，通讯延迟等因素)

（2）异步APRAM模型：

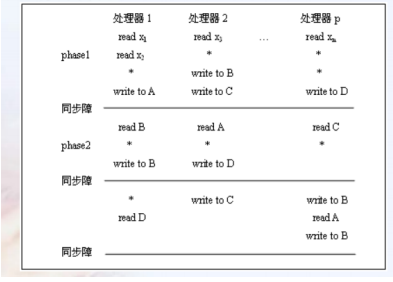
每个处理器有其局部存储器、局部时钟、局部程序；无全局时钟，各处理器异步执行；处理器通过SM进行通讯；处理器间依赖关系，需在并行程序中显式地加入同步路障。又称分相（Phase）PRAM或MIMD-SM。

指令类型
(1)全局读 (2)全局写 (3)局部操作 (4)同步



优点：易编程和分析算法时间复杂度
缺点：与现实相差元，并行算法非常有限，不适合MIMD-DM模型

计算过程
由同步障分开的全局相组成



（3） BSP模型：

“块”同步模型，是一种异步MIMD-DM模型，支持消息传递系统，块内异步并行，块间显式同步。

特点：
将CPU与选路器分开，强调计算任务与通信任务的分开，简化通信协议
采用路障方式；以硬件实现的全局同步；紧耦合同步式；
优点：强调了计算机和通讯的分离，提供了一个编程环境，易于程序复杂性分析
缺点：需要显式同步机制，限制至多h条消息的传递。

（4） logP模型：

由Culler(1993)年提出的，是一种分布存储的、点到 点通讯的多处理机模型，其中通讯由一组参数描述， 实行隐式同步。
优点：捕捉MPC的通讯瓶颈，隐藏并行机的网络拓扑、路由、协议，可以应用到共享存储、消息传递、数据并行的编程模型中；
缺点：难以算法描述、设计和分析

★比较BSP和logP

• BSP→LogP：BSP块同步→BSP子集同步→BSP进程对同步＝LogP

- BSP可以常数因子模拟LogP，LogP可以对数因子模拟BSP
- BSP＝LogP+Barriers－Overhead
- BSP提供了更方便的程设环境，LogP更好地利用了机器资源
- BSP似乎更简单、方便和符合结构化编程

4.静态互联网络节点度、对剖宽度

（考最简单静态网络的对刨宽度和节点度计算（只有2D网孔）

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。

对剖宽度（Bisection Width）：对分网络各半所必须移去的最少边数

简答3 5.超线性加速 什么是，什么情况会出现）

实际加速比相较于算出来的加速比更快，优于计算结果

例如，在某些并行搜索算法中，允许不同的处理器在不同的分支方向上同时搜索，当某一处理器一旦迅速地找到了解，它就向其余的处理器发出中止搜索的信号，这就会提前取消那些在串行算法中所作的无谓的搜索分支，从而出现的超线性加速。

举例：什么时候会出现超线性加速现象？又如：某一问题执行在大量处理器上，他的大多数据放在高速缓存中，总的计算时间趋于减少。这种高速缓存效应造成的计算时间下降补偿了由于通信等造成的额外开销时间，则可能造成超线性加速现象。

简答4：6.PRAM 模型分类（根据内存可分几类前面出现过了）

(1) PRAM-CRCW并发读并发写

- **CPRAM-CRCW(Common PRAM-CRCW)：仅允许写入相同数据**
- **PPRAM-CRCW(Priority PRAM-CRCW)：仅允许优先级最高的处理器写入**
- **APRAM-CRCW(Arbitrary PRAM-CRCW)：允许任意处理器自由写入**

(2) PRAM-CREW并发读互斥写

(3) PRAM-EREW互斥读互斥写

计算能力比较

- **PRAM-CRCW是最强的计算模型，PRAM-EREW可logp倍模拟PRAM-CREW和PRAM-CRCW TEREW≥TCREW≥TCRCW**

简答5：7.域分解、功能分解（了解概念、区别没整？？？）

1. 域分解

划分对象：划分的对象是数据，可以是算法的输入数据、中间处理数据和输出数据

概念：①将数据分解成大致相等的小数据片

②划分时考虑数据上的相应操作

③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

2. 功能分解

划分对象：划分的对象是计算，将计算划分成不同的任务，其出发点不同于域分解

概念：划分后，研究不同任务所需的数据。如果数据不相交，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

特点：功能分解是一种更深层次的分解

划分依据

①划分是否具有灵活性

②划分是否避免了冗余计算和存储

③划分任务尺寸是否大致相当

④任务数与问题尺寸是否成比例

⑤功能分解是否是一种更深层次的分解，是否合理

简答6.冉令强举例：带状划分、棋盘划分（了解概念） 区别

带状划分：就是将矩阵整行或整列地分成若干组，每组给指派一个处理器，也可将若干行或若干列指派给一个处理器，而且这些行和列可以是连续的，也可以是等距相间的，前者称为亏带状划分，后者称为循环带状划分。

棋盘划分：所谓棋盘划分就是将仿真划分成若干子方针，每个子方阵指派给一个处理器，此时任意处理器均不包含整行或整列。和带状划分类似，棋盘划分也可分为快棋盘划分和循环棋盘划分。

区别：1.从处理器的利用能力来看，棋盘划分比带状划分要好。2.和带状划分相比，棋盘划分可开发出更高的并行度。对于一个n*n的方阵，棋盘划分最多用n^2个处理器进行并行计算，但使用带状划分可用的处理器不能多于n个。3.在网孔上用同样多的处理器，棋盘划分的矩阵向量乘法比带状划分时要更快。如果p>n,则无法使用带状划分，而棋盘划分不受此限制。即使p≤n，棋盘划分也更优。4.棋盘划分的矩阵-向量乘的可扩放性也比带状划分好。

9.高速缓存一致性问题（同16.几种DSM系统实现方案）（考选择考填空考大题）

大多数使用基于目录的高速缓存一致性协议；保留SMP结构易于编程的优点，也改善常规SMP的可扩放性；CC-NUMA实际上是一个分布共享存储的DSM多处理机系统；它最显著的优点是程序员无需明确地在节点上分配数据，系统的硬件和软件开始时自动在各节点分配数据，在运行期间，高速缓存一致性硬件会自动地将数据迁移至要用到它的地方。

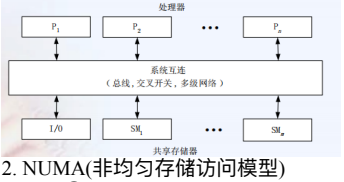
1. UMA(均匀存储访问模型)

特点：①物理存储器被所有处理器均匀共享

②所有处理器访问任何存储字取相同时间

③每台处理器可以带私有高速缓存

④外围设备也可以一定形式共享

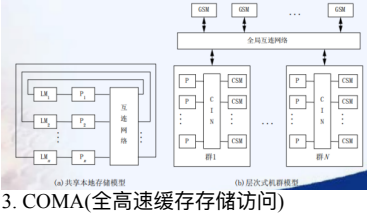


2. NUMA(非均匀存储访问模型)

特点：①所有本地存储器的集合组成了全局地址空间

②处理器访问存储器的时间不一样，LM或CSM较快，GSM较慢

③每台处理器照例可带私有高速缓存



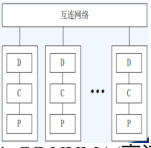
3. COMA(全高速缓存存储访问)

特点：①个处理器节点没有存储层次结构，全部高速缓存组成了全局地址空间

②利用高速缓存目录D进行远程高速缓存的访问

③COMA的高速缓存容量一般都大于2级高速缓存容量

④使用COMA时，数据开始时可以任意分配



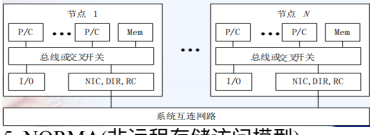
4. CC-NUMA(高速缓存一致性非均匀访问模型)

特点：①大多数使用基于目录的高速缓存一致性协议

②保留SMP结构易于编程的优点，也改善常规SMP的可扩放性

③CC-NUMA实际上时一个分布共享存储的DSM多处理机系统

④最显著的优点时程序员无需明确地在节点上分配数据



5. NORMA(非远程存储访问模型)

特点：①所有存储器私有，仅能由其处理器访问

②绝大多数NORMA不支持远程存储器的访问

10.两种开关技术
（简答：原理 特点 两种之间的区别）

1.存储转发（SF）技术

概念：消息被分为基本的传输单位—信包，每个信包都包含寻路信息，信包到达中间A时，A放入通信缓冲器中，然后通过选路算法算出下一个节点B，AB通道空闲时，并且B的缓冲器可用时A->B

传输时间 $t_{momm}(SF) = t_s + (mt_w + t_h)l$ ： tcomm(SF)=ts+(mt_w+th)l=O(ml)

缺点：1.每个结点必须对整个消息和信包缓冲2.网络延迟与发送消息的节点数成正比

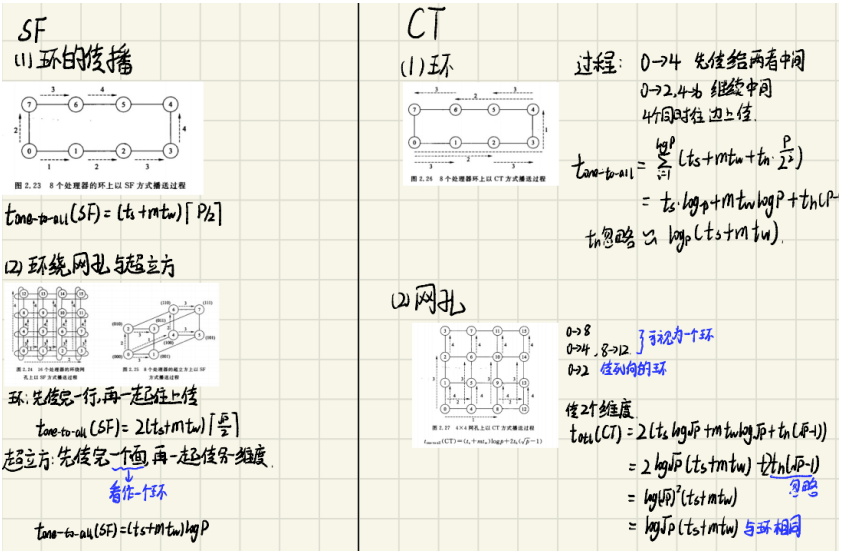
2.切通选路（CT）

概念：传递消息之前先建立一条从源节点到目的结点的物理通道，传递过程中一直占用，知道传输完毕才废弃，将信包进一步分成更小的片进行传输，同一信包中所有片一同以流水线方式穿越网络

传输时间： $t_{momm}(CT) = t_s + mt_w + lt_h$ O(m+l)

缺点：1.物理通道非共享2.传输过程中物理通道一直被占用

3.SF与CT的one-to-all传播



11.等效率函数曲线（给曲线解释分析哪条曲线性能更好，为什么）（直接看lzc的吧）

（给一个等效率函数曲线的图，分析哪个的性能好性能差，为什么那个好）（ppt没找到）

12.三种加速比性能定律（三种定律前提适用条件，哪种情况下用哪个定律（给情境）公式推导，三个定律相互关系、区别）

值大于1，并行速度>串行速度，相对于串行速度快多少倍

Amdahl定律：固定计算负载；Gustafson定律：适用可扩放问题；Sun和Ni定律-受限于存储器

Amdahl 定律：

计算负载是固定不变的，适用于实时应用问题。当问题的计算负载或规模固定时，我们必须通过增加处理器数目来降低计算时间；加速比受到算法中串行工作量的限制。意味着，随着处理器数量无限增大，并行系统所能达到的加速上限为 1/f。悲观的

推导：

$$T_s = fW + (1-f)W \qquad T_p = fW + \frac{(1-f)W}{p}$$

$$S_p = \frac{W}{fW + \frac{(1-f)W}{p}} = \frac{p}{pf + 1 - f} = \frac{1}{(p-1)f + 1} \xrightarrow{p\to\infty} \frac{1}{f}$$

Gustafson定律：

对于很多大型计算，精度要求很高，即在此类应用中精度是个关键因素，而计算时间是固定不变的。此时为了提高精度，必须加大计算量，相应地亦必须增多处理器数才能维持时间不变；表明：随着处理器数目的增加，串行执行部分f不再是并行算法的瓶颈。意味着，随着处理器数量增大，加速几乎与处理器数成比例的线性增加，串行比例f不再是程序的瓶颈。乐观。

Gustafson加速定律（1988）：

$$S'=\frac{W_S+pWp}{W_S+p\cdot Wp/p}=\frac{W_S+pWp}{W_S+W_P}$$

$$S' = f + p(1-f) = p + f(1-p) = p-f(p-1)$$

并行开销W_o：

$$S'=\frac{W_S+pW_P}{W_S+W_P+W_O}=\frac{f+p(1-f)}{1+W_O/W}$$

Sun 和 Ni定律：

充分利用存储空间等计算资源，尽量增大问题规模以产生更好/更精确的解。是Amdahl定律和Gustafson定律的推广。

- 设单机上的存储器容量为M，其工作负载W=fW+(1-f)W
  当并行系统有p个结点时，存储容量扩大了pM，用G(p)表示系统的存储容量增加p倍时工作负载的增加量。则存储容量扩大后的工作负载为W=fW+(1-f)G(p)W，所以存储受限的加速为

$$S''=\frac{fW+(1-f)G(p)W}{fW+(1-f)G(p)W/p}=\frac{f+(1-f)G(p)}{f+(1-f)G(p)/p}$$

并行开销W_o:

$$S''=\frac{fW+(1-f)WG(p)}{fW+(1-f)G(p)W/p+W_O}=\frac{f+(1-f)G(p)}{f+(1-f)G(p)/p+W_O/W}$$

- $G(p)=1$时，就是Amdahl加速定律
- $G(p)=p$变为$f+p(1-f)$，就是Gustafson加速定律
- $G(p)>p$时，相应于计算机负载比存储要求增加得快，此时Sun和Ni加速均比Amdahl的加速和Gustafson加速高

13.并行计算对云计算、大数据、人工智能的支撑作用（第0个ppt）

核心支撑： 算力（并行计算）+算法（并行算法）+数据（并行存储）
算力 = 并行体系结构（硬件）+并行程序
算法 = 并行分布式算法（思维）+并行编程
数据 = 分布式存储技术（数据组织）+存储系统

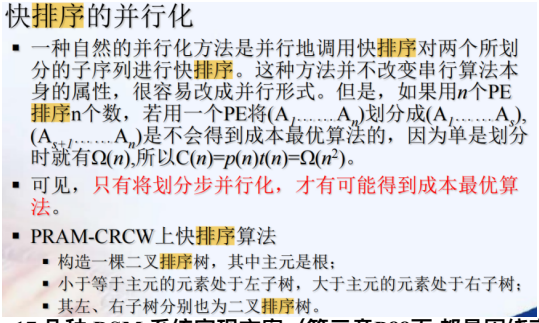
并行计算是为快速解决大型问题而进行沟通和协作的处理元素集合，并行计算是解决未来巨大计算需求的有效方法。

大量的计算问题本质上是并行的。但它们现有的应用程序是为单处理器系统设计的。它们需要并行化。

14.异步 PRAM 模型的同步障(为什么会产生同步障、同步障是做什么的）看lzc的）

15.矩阵和向量相乘的并行化（有哪些方法 具体是怎么并行化的 步骤是什么）

16.快速排序算法的并行化（快排算法的串行算法是什么样的 有哪些步骤可以并行化 有哪些问题是需要解决后才能实现并行化的 具体是怎么解决的 最后并行化是怎么来的并实现的 文字叙述 不要求代码）



17.几种 DSM 系统实现方案（第三章P88页 都是围绕高速缓存一致性问题展开 每种特性 相互之间有什么区别 考选择考填空考大题）

(1) CC-NUMA结构(高速缓存一致性非均匀存储结构系统)
(2) NCC-NUMA结构(非高速缓存一致性非均匀存储结构系统)
(3) COMA结构(全高速缓存存储结构系统)
(4) 共享虚拟存储系统
(5) 前四种的结合

18.并行计算模型的演化过程和发展路线（第五章167页小结和导读 要结合第三条模型随着主流并行机体系结构变化的演化过程。

共享存储体系结构->分布存储体系结构->分布共享存储体系结构

第一代共享存储模型;它适应于共享存储的SIMD和MIMD并行机;以 CPU计算为中心，将CPU计算、数据存储和通信传输等不作区分，并将所有的运算都归一化为 CPU 的单步时间操作，计算过程中可锁步（自动同步）或异步进行;代表性模型主要有 PRAM 和APRAM 等。

第二代分布存储模型:它适应于分布存储的 MIMD并行机;以数据通信为中心，将 CPU计算操作和 I/O通信操作区分开来，认为通信时间是算法运行时间的主体;计算过程中可整体大同步或个体异步进行;代表性模型主要有BSP和LogP 等。

第三代分布共享存储模型;它适应于分布共享的 MIMD并行机;以数据访问为中心，将CPU计算和数据访问操作严格区分开来，认为访问位于不同层次存储器中数据的时间是算法运行时间的主体;计算过程中可同步或异步进行;代表性模型主要有 DRAM（h）和 Memory-LogP等。

模型沿着不断完善强化单一模型功能的路线而发展。

起初模型中仅有反映 CPU特性的单一参数，然后模型中又考虑了反映通信网络特性的一些参数（如延迟、带宽等），近来模型中又引入了考虑不同存储器（如访问寄存器、高速缓存、主存和外存等）访问特性的一组参数。

描述机器特性的参数越来越多，使得模型的功能越来越强、准确性越来越好，但功能太复杂和参数过多致使单一模型描述困难和难以求解其成本函数，最终导致单一模型难以操作和不实用。所以近年来，提出分层研究方法。

并行计算模型应按并行算法设计模型并行程序设计模型并行程序执行模型 各个层次的职能分工明确和目标单一，可将单一模型中的各种功能按要求分配到模型的不同层次中去，令其各负其责，缓解了单一模型的精确性和可使用性之间的矛盾，从而易于设计和实现以及模型更加实用

2. 习题

课堂作业

1.分析一下并行计算结构发展过程，经历哪些阶段，互相之间什么关系

模型随着主流并行机体系结构变化的演化过程。

共享存储体系结构->分布存储体系结构->分布共享存储体系结构

第一代共享存储模型;它适应于共享存储的SIMD和MIMD并行机;以 CPU计算为中心，将CPU计算、数据存储和通信传输等不作区分，并将所有的运算都归一化为 CPU 的单步时间操作，计算过程中可锁步（自动同步）或异步进行;代表性模型主要有 PRAM 和APRAM 等。

第二代分布存储模型;它适应于分布存储的 MIMD并行机;以数据通信为中心，将 CPU计算操作和 I/O通信操作区分开来，认为通信时间是算法运行时间的主体;计算过程中可整体大同步或个体异步进行;代表性模型主要有BSP和LogP 等。

第三代分布共享存储模型;它适应于分布共享的 MIMD并行机;以数据访问为中心，将CPU计算和数据访问操作严格区分开来，认为访问位于不同层次存储器中数据的时间是算法运行时间的主体;计算过程中可同步或异步进行;代表性模型主要有 DRAM（h）和 Memory-LogP等。

模型沿着不断完善强化单一模型功能的路线而发展。

起初模型中仅有反映 CPU特性的单一参数，然后模型中又考虑了反映通信网络特性的一些参数（如延迟、带宽等），近来模型中又引入了考虑不同存储器（如访问寄存器、高速缓存、主存和外存等）访问特性的一组参数。

描述机器特性的参数越来越多，使得模型的功能越来越强、准确性越来越好，但功能太复杂和参数过多致使单一模型描述困难和难以求解其成本函数，最终导致单一模型难以操作和不实用。所以近年来，提出分层研究方法。

并行计算模型应按并行算法设计模型并行程序设计模型并行程序执行模型 各个层次的职能分工明确和目标单一，可将单一模型中的各种功能按要求分配到模型的不同层次中去，令其各负其责，缓解了单一模型的精确性和可使用性之间的矛盾，从而易于设计和实现以及模型更加实用。

2. 对比一下带状划分和棋盘划分有什么区别

1.从处理器的利用能力来看，棋盘划分比带状划分要好。2.和带状划分相比，棋盘划分可开发出更高的并行度。对于一个n*n的方阵，棋盘划分最多用n^2个处理器进行并行计算，但使用带状划分可用的处理器不能多于n个。3.在网孔上用同样多的处理器，棋盘划分的矩阵向量乘法比带状划分时要更快。如果p>n,则无法使用带状划分，而棋盘划分不受此限制。即使p≤n，棋盘划分也更优。4.棋盘划分的矩阵-向量乘的可扩放性也比带状划分好。

3. **对比方根划分和对数划分**

方根划分是将第i√n个元素作为划分元素，将序列分成若干段，然后分段处理。这种划分技术使得算法达到了很好的时间界，可以在理想时间复杂度内完成序列的归并。对数划分时取第ilogn个元素作为划分元素分段然后分段处理。在方根划分中，如有AB两序列，为确定A中划分元素在B中的全局位序，划分元素须在各段之间全局比较。而对数划分在选取A序列中划分元素时已考虑到A在B中的全局位序，不必对划分元素施行段间的全局比较，可直接对按划分元素断开的各段组两两进行归并，完成两原序列的归并。

4. 简述一下并行计算机几种模型发展历程

（同1）

5. 并行分布计算的作用，在哪些方面有应用，什么影响

并行计算（parallel computing）是指，在并行机上，将一个应用分解成多个子任务，分配给不同的处理器，各个处理器之间相互协同，并行地执行子任务，从而达到加速求解速度，或者求解应用问题规模的目的

开放题：