一. 名词解释.　07, 08 部分作答

1. SIMD: Single Instruction Stream Multiple Data Stream. 即单指令流多数据流. 弗林分类法对计算机分类的一种计算机系统结构. 通过一个控制单元同时控制多个数据单元运动.

2. SMP: Symmetric Multiprocessor, 即一种共享存储的并行机类型, 称为对称多处理机. 一般利用系统总线作为互连网络实现通信.

3. UMA: Uniform Memory Access 即均匀存储访问模型, 是并行计算机的一种主要访存模型. 指所有处理器访问任何存取字取相同的时间.

4. DSM: Distributed Shared Memory. 即分布式共享存储多处理机. 一种并行计算机系统类型通常也称为紧密耦合多处理机, 具有一个所有处理器都可以一致访问的全局物理内存.

5. MTTF: Mean Time To Fail 即平均无故障时间. 用来衡量系统可靠性的度量, 系指系统失效前平均正常运行的时间.

6. TFLops: Trillion Floating Point Operations Per Second, 即每秒万亿次浮点运算. 一种用来度量并行机器性能的度量单位.

7. SSI: Single System Image, 即单一系统映像, 是机群的一个重要特征, 使用它使得机群在使用控制和维护上像是一个工作站.

8. SAF: 即 Service Availability Forum 也称为 SA Forum, 即服务可用性论坛. 一个致力于定义一组用于电信设备和其他商业设备管理公共接口的组织, 主要定义有 HPI 接口等.

9. MIN: Multistage Interconnection Network. 即多级互连网络, 是为了构筑大型开关网络, 将单级交叉开关级联起来形成的一种互连网络. 用于 MIMD 和 SIMD 计算机设计中.

10. COW: Cluster of Workstations 即机群, 一种新型的并行机体系结构. 主要为将一群工作站或高档微机用某种结构的互连网络互连起来, 实现高效并行计算. P18

07
08

11. MPP: Massively Parallel Processor 即大规模并行处理机. 一种并行计算机系统, 是指由成百上千甚至近万个处理器组成的大规模计算机系统.

12. PVP: Parallel Vector Processor 即并行向量处理机, 一种自路优称一时的并行计算机, 一般包含少量的高性能专用设计定制的向量处理器且使用专门设计的高带宽交叉开关网络.

13. GFLOPS: Giga (Billion) Floating Point Operations Per Second. 即每秒十亿次浮点运算一种用来衡量并行机器性能的度量单位.

14. Hypercube: 即超立方. 一种普遍正交的静态互连网络, 是n维网格和k元n立方的特例.

15. Cut-Through: 并行计算机互连网络中的一种选路方式. 即切通选路, 以虫蚀选路为代表. 将信包进一步分成更小的片进行传输. 分成更小的片进行传输

MTTR 是系统服务性的度量, 表示系统正常工作的时间

一、问答题

1. 请简单介绍并行计算机访存模型。07-二-1（08-二-1）

答：并行计算机访存模型主要包括如下模型。

① UMA 即均匀存储访问模型，它指所有处理器访问任何存取储单元取相同的时间。

② NUMA 模型即非均匀存储访问模型，它指处理器访问不同的存储器时间不一样。

③ COMA 模型即高速缓存存储访问，是 NUMA 模型的一种特例，没有存储层次，而是由全部高速缓存构成了全局地址空间。

④ CC-NUMA 模型，即高速缓存一致性非均匀存储访问模型。

⑤ NORMA 模型，即非远程存储访问模型，它指所有存储器都私有的访存模型。

⑥ NCC-NUMA：即高速缓存不一致非均匀存储访问模型。

2. 请比较 Amdahl 定律和 Gustafson 定律。07-二-2

答：Amdahl 定律和 Gustafson 定律都是用于评价加速比性能的定律，其中 Amdahl 定律适用于固定计算负载的加速比，Gustafson 定律适用于可扩放问题的加速比计算。

单从结果上看，两者似乎是对立的，但实际是统一的，对立结果只是由于两者的前提条件不一定。

其中 Amdahl 定律有 $S=\frac{p}{1+f(p-1)}$ 当 $p\to\infty$ $S=1/f$ 意味着处理器数目随着无限增大，并行系统所能到达的加速上限为 $1/f$，对并行系统只有悲观的作用。

对于 Gustafson 定律有 $S=p-f(p-1)$，即 p 充分大，S 与 p 几乎成线性，其意味随着处理器数量的增加，加速比几乎与处理器成比例增加，f 不再是瓶颈，对并行系统发展是个乐观结论。

本质上是因为 Amdahl 定律为固定负载，而 Gustafson 为可扩放，即相对于 Amdahl 定律来说 Gustafson 随着处理器数量增加，问题规模扩大，实际上是相对于 Amdahl 定律中负载不变来说，增加了并行负载部分的比例，即降低了 f，因此，并不矛盾。

3. 请比较描述可扩放性评价中的三种评价标准。08-二-2

答：可扩放性评价标准主要包括：

① 等效率度量标准：即在保持效率 $E=\frac{S}{p}$ 不变的前提下，研究问题规模 W 如何随着处理器数 p 而变化，即为了保证 E 不变，需要在 p 增大时相应的增加多少问题规模。

② 等平均速度度量标准：是在效率 E 不变的前提下，用平均速度的比值来描述随着处理器数 p 的增加需要增加的工作量 W。

③ 等速度度量标准：是在保持平均速度 $V=\frac{W}{pT}$ 不变的前提下，研究处理器个数增大时应相应的增加多少工作量。

三种评判可扩放性度量标准的基本出发点，都是抓住了影响算法可扩放性的基本参数 T_0，只是①采用解析计算方法得到 T_0，②将 T_0 隐含在测量的执行时间中，③是在保持 E 为恒值时通过调节 W 与 P 来测量并行和串行执行时间，最终通过平均速度反映出 T_0。

08-二-3

[illegible]

答：任务划分主要方法有数据划分和计算划分，其主要思想如下：

① 域分解，即数据划分。其划分的对象为数据。首先是分解与问题相关的数据，如果可能的话，应使这些小的数据片尽可能大致相当。其次再将每个计算关联到它所操作的数据上。如在一个三维网格上，在各格点上计算都是重复执行的，因此可以进行分区分解，在X,Y,Z维上都可以。如进行三维分解，即每一个格点定义一个计算任务，每个任务维护与其格点相关的各种数据，并负责计算，以修改状态。

② 计算划分，即任务分解或功能分解。其首先关注于被执行的计算上，而不是计算所需数据。如果计算划分成功，再继续研究计算所需的数据。如果数据基本不相交，则意味着划分成功；否则，应该考虑数据分解。如搜索树可以进行功能分解，开始时根生成一个任务，对其评价后若不是解，就生成一棵搜索子树。整个搜索过程可以波前形式逐级向树叶推进。

5. 请给出二维网格最小北最后、最小负优先路由算法。

08-二-4　07-二-3

最小北最后算法

```
输入：(XCurrent, Ycurrent)和目的(Xdest, Ydest)
输出：选择的输出通道 Channel
过程  Xoffset = Xdest - Xcurrent;
      Yoffset = Ydest - Ycurrent;
 if Xoffset > 0 and Yoffset < 0 then
     Chanel = Select(X+, Y-);
 endif
 if Xoffset < 0 and Yoffset < 0 then
     Channe = Select(X-, Y-);
 endif
 if Xoffset = 0 and Yoffset < 0
     channel = Y-
 endif
 if Xoffset > 0 and Yoffset >= 0
     channel = X+
 endif
 if Xoffset < 0 and Yoffset >= 0
     channel = X-
 if Xoffset = 0 and Yoffset > 0
     chanel = Y+
 endif
 if Xoffset = 0 and Yoffset = 0
     Channel = Internal
 endif
```

最小负优先算法

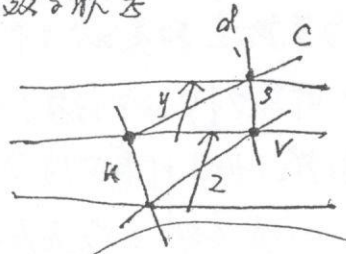
```
if Xoffset < 0 and Yoffset < 0
    channel = Select(X-, Y-)
endif
if Xoffset < 0 and Yoffset >= 0
    Channel = X-
endif
if Xoffset >= 0 and Yoffset < 0
    channel = Y-
endif
if Xoffset >= 0 and Yoffset > 0
    channel = select(X+, Y+)
endif
if Xoffset > 0 and Yoffset = 0
    channel = X+
endif
if Xoffset = 0 and Yoffset > 0
    Channel = Y+
endif
if Xoffset = 0 and Yoffset = 0
    Channel = Internal
```

08-二-8

答：故障恢复有向后恢复和向前恢复两种方案：

①向后恢复：周期性地将一个一致的状态（检查点）保存至稳定的外存中。在发生失效后，系统重配置以隔离失效组件，恢复至前一个检查点以继续正常操作，也称为回卷。

②向前恢复：如果执行时间很关键，例如在实时系统中，不允许有回卷时间，则此时系统不回卷至前一个检查点，而是利用失效诊断信息重建一个有效的状态



如果进程之间不存在一个进程的检查点已接收到的消息，而为一个进程的检查点还未发出消息，则是一致的。如图，c是不一致的。

07-二-5

11. 利用LL和SC原语指令，实现一个Array-based

实现 Compare & Swap

```
try: LL reg1, location
     bnz reg1, try
     sub reg3, reg2, reg1
     bnz reg3, try
     SC location reg4
     beqz reg4 try
     ret
```

LL/SC实现 Test & Set

```
lock: LL, reg1, location
      bnz reg1, try
      mov reg2, #1
      SC location reg2
      beqz reg2, try
      ret
unlock: st location, #0
        ret
```

实现 Array-based

```
Lock: LL reg1, next->available
      addi reg2, reg1, #1
      bnz reg1, lock
      addi, reg2, reg1, #1
      SC next->available, reg2
      beqz reg2, lock
wait: bnz array[reg2], wait
unlock:
      addi, reg3, reg2, #1
      st array[reg3], #0
      ret
```

st前所用标号为

LL/SC实现 Ticket Lock

next_ticket

```
Lock: LL reg1, location
      addi reg1, reg1, #1
      SC location reg1
      beqz reg1, lock
wait: sub reg2, now_serving, reg1
      bnz reg2, wait
unlock: addi now_serving, now_serving, #1
        ret
```

bnz reg1, lock

fetch & inc

```
Lock: LL reg1, location
      bnz reg1,
      addi reg1, reg1, #1
      SC location reg1
      beqz reg1, lock
      ret
```

6. 请简单比较基于侦听的高速缓存一致性协议和基于目录的高速缓存一致性协议。08-二-6 (07-二-6)

答：基于侦听的高速缓存一致性协议通过接收处理器请求和侦听总线上的事务作为输入，并对这些事务做出对应的操作以保证高速缓存一致。P221

而基于目录的高速缓存一致性协议则是通过维护一个全局目录，每次在需要通过一致性命令来维护高速缓存一致性时利用全局目录点把一致性命令发给相关节点从而实现一致性协议。

基于侦听的协议适用基于总线的SMP机器，利用总线的事务对所有控制器可见和次序一致的特点实现，但不适用于大型可扩展网络中，因为在大型系统中总线的带宽成为了限制，特别是在网络进行广播时，代价太大。而基于目录的协议则适用于大型机器中，可以通过把目录分布到各个节点上从而防止其成为系统瓶颈。

7. 请回答何为机群中的单一系统映像以及它主要包括哪些服务。08-二-8

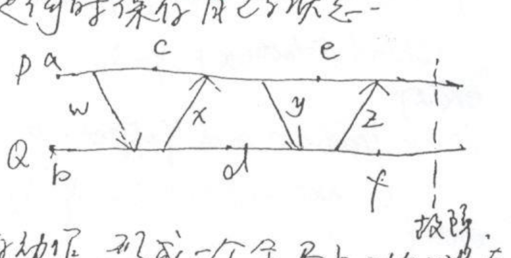
答：机群中的单一映像系统主要指对机群实现单一系统、单一控制、对称性和位置透明化个特征，即尽管系统有多个处理器，用户仍然可以作为一个单一系统使用机群，在逻辑上，最终用户或系统用户使用的服务都来自只有惟一接口的同一地方，且用户可以从任何一个节点上来获得服务，不需要了解具体运行服务的物理设备位置。

其服务有：①单一入口点 ②单一文件层次 ③单一输入/输出 ④单一管理和控制点 ⑤单一网络 ⑥单一存储空间 ⑦单一作业管理系统 ⑧单一用户界面 ⑨单一进程空间。

8. 请回答为何执行检查点操作会产生多米诺效应？如何解决？07-二-7

答：产生多米诺效应的主要原因是因为系统中每个进程单独决定何时保存自己的状态。

如图：假定系统发生故障，P先回e，由于需要通信z从进程Q获得一到至发消息，Q回卷到d，同理，P又回卷到c，最后P、Q到a、b，即产生多米诺效应。



解决方法一是采用协同式的检查点恢复技术，进程协调保存动作，形成一个全局上一致的状态；二是仍然使用检查点独立，但增加一个日志消息，即进程不仅都独立地保存它们的局部检查点，而且同时将消息封存在日志中。

9. 请比较描述SMP中不同的锁机制 08-二-7

答：SMP中的锁机制主要包括以下几个方面：

(1)即支持原子执行的原子交换指令，如Lock和Unlock及交换指令Swap，同时还有Test&Set和Fetch&Op等几种原子指令。这些原子交换指令大都可以通过处理器支持的一对特殊指令LL/SC来实现。

(2)对简单锁改进提高性能的锁，如Test&Set lock with backoff和Test-and-Test&Set锁

(3)还有支持更加先进功能的锁，如Ticket Lock等锁和基于数组的Array-based Lock，其为以实现更高级功能的锁。

(4)全硬件实现的锁，每一根专门用于锁的线可以用来实现一个锁。