1. **请简述Flynn分类法将计算机系统结构分成哪四类。**
2. 单指令流，单数据流
3. 单指令流，多数据流
4. 多指令流，单数据流
5. 多指令流，多数据流
6. **请简述程序局部性原理。**

局部性原理：程序执行时所访问的存储器地址分布不是随机的，而是相对地簇聚。

（1）程序的时间局部性：程序即将用到的信息很可能就是目前正在使用的信息

（2）程序的空间局部性：程序即将用到的信息很可能与目前正在使用的信息在空间上相邻或者临近。

1. **请简述Amdahl定律。**

Amdahl定律：加速某部件执行速度所能获得的系统性能加速比，受限于该部件的执行时间占系统中总执行时间的百分比

加速比是衡量系统性能的改进多少的一个指标。

加速比== =改进前的总执行时间\*{（1-可改进比例）+}

加速比依赖于两个因素：可改进比例和部件加速比

1. **影响CPU时间的因素包括什么？（提示：从CPU公式入手，考虑3个参数的影响因素）**

CPU时间 = IC \* CPI \* 时钟周期时间

因素：

(1)时钟周期时间：取决于硬件实现技术和计算机组成

(2)CPI：取决于计算机组成和指令集结构

(3)IC：取决于指令集结构和编译技术

1. **请简述冯诺依曼体系结构的特点。**

计算机以运算器为中心

在存储器中，指令和数据同等对待

存储器是按地址访问、按顺序线性编址的一维结构，每个单元的位数是固定的

指令的执行是顺序的，即一般是按照指令在存储器中存放的顺序执行

指令由操作数和地址码组成

指令和数据均以二进制编码表示，采用二进制计算

1. **请简要说明提高计算机系统并行性的3种技术途径，并分别从单机和多机系统的角度举例。**

提高计算机系统并行性的3种技术途径：时间重叠、空间重叠、资源重复

时间重叠：引入时间因素，让多个处理过程在时间上错开，轮流重叠使用同一硬件设备的各个部件，加快硬件周转而赢得时间。

单处理机中（部件功能专用化）：流水线

多处理机中（处理机专用化）：专用外围处理机、专用处理机、异构型多 处理机系统

空间重叠：引入空间因素，通过重复设置硬件资源，以量取胜，大幅度提高计算机系统性能。

单处理机中（重复设置功能部件）：多存储并行、数据/指令独立存储体、重复设置运算部件

多处理机中（重复设置处理机）：容错系统、同构性多处 理机系统

资源共享：一种软件方法，使多个任务按一定时间顺序轮流使用一套硬件设备。

单处理机中：分时系统，虚拟机

多处理机中：分布式系统

1. **请分析影响平均访存时间的因素有哪些？针对每个因素，各举出1种改进方法。**

影响平均访存时间的因素：不命中率、不命中开销、Cache命中时间

（1）降低不命中率：

增加块的大小——降低强制性不命中

增加Cache容量——降低容量不命中

提高相联度——降低冲突不命中

伪相联Cache、硬件预取、编译器预取、编译器优化、牺牲Cache

（2）减小不命中开销：采用多级Cache（两级）、让读不命中优先于写、写缓冲合并、请求字处理、非阻塞Cache技术

（3）降低命中时间：使用小容量、结构简单的Cache、虚拟Cache、Cache访问流水话、跟踪Cache

8. **请简述Cache的3C失效。**

三种类型的不命中：

（1）强制性不命中（Compulsory miss）:当第一次访问一个块时，该块不在Cache中，需从下一级存储器中调入Cache，这就是强制性不命中。

（2）容量不命中（Capacity miss）:如果程序执行时所需的块不能全部调入Cache中，则当某些块被替换后，若又被重新访问，就会发生不命中，这种不命中称为容量不命中。

（3）冲突不命中（Conflict miss）：在组相联或者直接映像Cache中，若太多的块映射到同一组（块）中，则会出现该组中某个块被别的块替换（即使别的组或块有空闲位置），然后又被重新访问的情况，这就是发生了冲突不命中。

9. **请针对3C失效的每种失效给出一种降低失效率的方法，并分别分析该种方法的缺点**。

（1）针对强制不命中 -- 增加块大小

增加块大小会增加不命中开销；会造成冲突不命中增加

（2）针对容量不命中 -- 增加Cache容量

增加成本；可能增加命中时间

（3）针对冲突不命中 -- 提高相联度

提高相联度是以增加命中时间为代价

10.**请简述伪相联的基本思想。**

基本思想：在逻辑上把直接映像Cache的空间上下平分为两个区。

对于任何一次访问，伪相联Cache先按直接映像Cache的方式去处理：若命中，则其访问过程与直接映像Cache的情况一样；若不命中，则再到另一区相应的位置去查找，若找到，则发生了伪命中，否则就只好访问下一级存储器。

11.**请例举一种编译器优化的方法，并举例说明**。

（1）数组合并

将本来相互独立的多个数字合并成为一个复合数组，以提高访问它 们的局部性。

（2）内外循环交换

（3）循环融合

将若干个独立的循环融合为单个的循环，这些循环访问同样的数 组，对相同的数据作不同的运算。这样能使得读入Cache得数据在被替换出去之前，能得到反复的作用。

（4）分块

12.**请说明流水线中有哪三种相关？分别会引起哪种流水线中的冲突（冒险）？**

（1）数据相关：数据冲突中的写后读

（2）名相关：数据冲突中的读后写、写后写

（3）控制相关：控制冲突（冒险）

13.**请说明什么是静态调度？什么是动态调度？动态调度的优点是什么？**

静态调度：依靠编译器对代码进行调度，也就是在代码被执行之前进行调度，通过把相关的指令拉开距离来减少可能产生的停顿

动态调度：在程序的执行中，依靠专门硬件对代码进行调度，减少数据相关导致的停顿。

动态调度的优点：相对于静态指令调度，动态指令调度是在指令地执行过程中进行调度，使得无关的指令得以先执行，减少阻塞，能够处理一些在编译时情况不明的相关（如存储器访问的相关）；能够是本来是面向某一流水线优化编译的代码在其他的流水线（动态调度）上也能高效地执行。

14.**请简述Tomasulo算法的基本思想。**

核心思想：记录和检测指令相关，操作数一定就绪就立即执行，把发生 RAW冲突的可能性减少到最小；通过寄存器换名来消除WAR冲突和WAW冲突。

15.**请说明什么是动态分支预测？有何优点？**

动态分支预测技术：通过硬件技术，在程序执行时根据每一条转移指令过去的转移历史记录来预测下一次转移的方向。通过提前预测分支方向，减少或消除控制相关导致的流水线停顿。

优点：

1、根据程序的执行过程动态地改变转移的预测方向，因此有更好的准确度和适应性。

2、程序每次执行时，可能预测的分支方向与前次相同或不同。

16.**请简述分支历史表BHT的基本思想。**

基本思想：用BHT来记录分支指令最近一次或几次的执行情况（成功或不成功），并据此进行预测。

17.**请简述分支目标缓冲器BTB的基本思想。**

基本思想：将分支成功的分支指令的地址和它的分支目标地址都放到一个缓冲区保存起来，缓冲区以分支指令的地址作为标识。这个缓冲器就是分支目标缓冲器（BTB）.

当程序执行遇到分支指令时，先再分支目标缓冲器进行查询，并据结果进行分支方向预测。

18.**请简述基于硬件的前瞻算法基本思想。**

基本思想：对分支指令的结果进行猜测，并假设这个猜测总是对的，然后按这个猜测结果继续取出、流出和执行后续的指令。只是执行指令的结果不是写回到寄存器或存储器，而是放到一个称为ROB的缓冲器中。等到相应的指令得到“确认”之后，才将结果写入寄存器或存储器。

* 动态分支预测：用来选择后续执行的指令
* 在控制相关的结果尚未出来之前，前瞻地执行后续指令
* 用动态调度对基本块的各种组合进行跨基本块的调度

19.**请简述向量体系结构和GPU体系结构的差异。**

1、向量体系结构：

“窄而深”，指令流水线深，ALU宽度窄，单次指令流水后能处理更多数据，掩盖不必要的流水线时间

2、GPU：

“宽而浅”，指令流水线浅，ALU宽度宽，流水本身比较简单，直接对更多的数据进行并行运算，同一时刻处理更多数据。

20.**请简述GPU和CPU在设计理念上的差异性。**

CPU是多核的，可以有几个，但是GPU是众核，可以达到几千几万个之多；CPU主要是控制，而GPU主要是运算：GPU将芯片中更多的晶体用于计算，大大提高了运算效率；CPU强在快速的指令级并行，快速的Cache访问，具有较少的Cache访问延迟，而GPU强在并行运算的能力，GPU距内存较远，因此访问延迟较高，但是访问带宽较高，所以又称为“高通量计算”。

**CPU被设计用于通用计算，因此具有复杂的控制单元和存储单元；**

**GPU用于处理计算性强而逻辑性弱的数据，有更多的计算单元和超 长流水。**

21.请简述GPU各个层次组件间的相似性。

22.**请简述GPGPU虚拟化的思想。**

在硬件上把多个SM分割成多个切片，软件上为每个用户分配一个切片，使得每个用户都能满负荷使用SM，每个用户感觉自己拥有了一个GPU。

GPGPU虚拟化是指将GPU（图形处理器）资源虚拟化并提供给多个虚拟机或应用程序使用的技术，其思想是利用GPU的并行计算能力，在虚拟环境中实现高效的图形渲染和并行计算。

23.**请简述向量长度寄存器和向量屏蔽寄存器的作用。**

（1）向量长度寄存器：将软件层程序中实际向量长度N与硬件层向量寄 存器中的元素数目64相匹配

（2）向量屏蔽寄存器：控制向量的掩码和选择，可以用于数据过滤，通过屏蔽位来过滤掉不需要的数据，还可以控制向量控制流。当向量长度小于64，或在条件语句控制下，对向量某些元素进行单独运算时使用。

24.**请简述指令编队的思想。**

将几条能在同一个时钟周期内一起开始执行的向量指令集合成为编队，同一个编队中的向量指令之间不存在结构冲突和数据冲突，或者存在数据冲突但是可以链接。

25.**请简述链接技术的思想。**

当两条指令出现“写后读”相关时，若它们不存在功能部件冲突和向量寄存器(源或目的) 冲突，就有可能把它们所用的功能部件头尾相接，形成一个链接(长)流水线，进行流水处理。

26.**请简述分段开采技术的思想。**

当向量的长度N大于向量寄存器的长度n时，必须把长向量N分成长度 固定为n的段，然后循环分段处理，每一次循环只处理一个向量段。这种技术称为分段开采技术。

27.**请在PVP、SMP、MPP、DSM和COW中任选一种，简要描述其特点。**

PVP：并行向量处理机。包含了少量的高性能专门设计定制的向量处理机VP；向量处理机VP和共享存储模块通过高带宽的交叉开关网络互连；通常不使用高速缓存，而是使用大量的向量寄存器和指令缓冲器。

28.**什么是多处理机Cache一致性问题？**

允许共享数据进入Cache，就有可能出现多个处理器的Cache中都有同一存储块的副本，当其中某个处理器对其Cache中的数据进行修改后，就会使得Cache中的数据与其他Cache中的数据不一致。

29.**请简述监听式协议的原理。**

每个Cache块除了包含物理存储器中块的数据拷贝之外，也保存着各个块的共享状态信息。Cache通常连载共享存储器的总线上，当某个Cache需要访问存储器时，它会把请求放到总线上广播出去，其他各个Cache控制器通过监听总线来判断它们是否有总线上请求的数据块，如果有，则进行相应的操作。

30.**请简述目录式协议的原理。**

目录：一种集中的数据结构。对于存储器中的每一个可以调入Cache的 数据块，在目录中设置一条目录项，用于记录该块的状态以及哪些Cache中有副本等相关信息。这使得任何一个数据块，都可以快速地在唯一的一个位置中找到相关的信息，这使一致性协议避免了广播操作。

**在物理存储器中，数据块的存储信息被存放在一个被称为目录的地方。目录协议根据该项目的信息和当前要进行的访问操作，依次向对于cache发送控制信息，并完成对目录项信息的修改。此外，还要向请求处理器发生响应信息。**

31.**请比较说明写作废协议和写更新协议性能上的差别。**

写作废协议：在处理器对一个数据块写入前，保证其对数据块唯一访问权（作废其他副本）；

写更新协议：处理器对一个数据块进行修改，修改信息通过广播使其他含有该块副本的cache更新。

性能差异：

1、在对同一个数据进行多次写操作而中间无读操作的情况下，写更新 协议需要进行多次写广播操作，而写作废协议只需一次作废操作。

2、在对同一个Cache块的多个字进行写操作的情况下，写更新协议对于每一个写操作都要进行一次广播，而写作废协议尽在对该块的第一次写时进行作废操作即可。

3、写作废是针对Cache块进行操作，而写更新则是针对字（字节）进行。写更新协议的延迟时间较小。

32.**请解释，在目录式协议中，什么是本地节点、宿主节点、远程节点和共享集合？**

（1）本地节点：发出访问请求的节点

（2）宿主节点：包含所访问的存储单元及其目录项的节点

（3）远程节点：存放有对应地址的存取器块及其目录项的节点，可以和宿主节点是同一个节点，也可以不同。

（4）共享集合；由位向量指定的处理机的集合成为共享集S。共享集S中包含所有具有该数据块的副本的处理机。

33.请简述目录式协议中，目录的三种结构。

1、全映像目录：每个目录项都包含一个N位（N为处理机的个数）的位向量，其每一位对应于一个处理机

2、有限映像目录：采用位数固定的目录项目，限制同一数据块在所有Cache中的副本总数

3、链式目录：用一个目录指针链表来表示共享集合，当一个数据块的副本数量增加或减少时，对应的链表的长度增加或减少。

