一、名词解释（6题，每题5分）

1、**ISA**

Instruction Set Architecture的缩写，指令集架构，是计算机体系结构中与程序设计有关的部分，包含了基本数据类型，指令集，寄存器，寻址模式，存储体系，中断，异常处理以及外部I/O。指令集架构包含一系列的opcode即操作码（机器语言），以及由特定处理器执行的基本命令

**2**、**MicroArchitecture**

微架构，指令集架构可以在不同的微架构上运行，微架构包含处理器内部的构成以及这些构成起来的部分如何运行指令集架构。

**3**、资源冲突

因硬件资源满足不了指令重叠执行的要求而发生的冲突。有些流水线处理机只有一个存储器，将数据和指令放在一起，访存指令会导致访存冲突

**4**、数据冲突

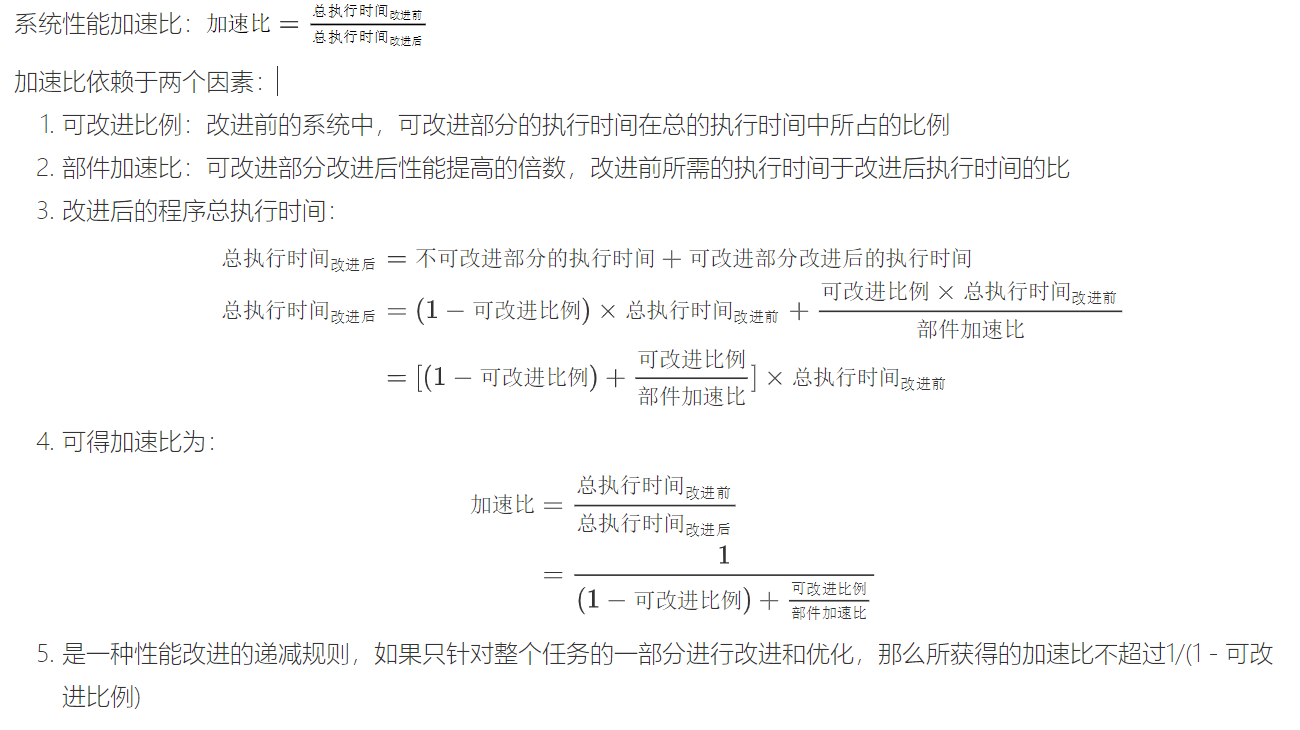
当指令在流水线中重叠执行时，因需要用到前面指令的执行结果而发生的冲突，有写后读冲突RAW，写后写冲突WAW，读后写冲突WAR

**5**、控制冲突

流水线遇到分支指令和其他会改变PC值的指令所引起的冲突

**6**、**Amdahl**定律

系统中某一部件由于采用某种更快的执行方式后整个系统性能的提高与这种执行方式的使用频率或占总执行时间的比例有关



**7**、加速比

不就上面那个？？？

**8**、非冯**·**诺依曼计算机

脱离了冯·诺依曼结构原有模式的计算机，例如光子计算机（光处理器利用光的高速和无干扰性，使用光学元件构成处理器，尚在研发中），并行计算机、数据流计算机以及量子计算机等

**9**、静态流水线、动态流水线、单功能流水线、多功能流水线

**静态流水线：**在同一时间内，多功能流水线中的各段只能按同一种功能的连接方式工作，只有输入为一串相同的运算任务时，流水的效率才得到充分的发挥

**动态流水线：**在同一时间内，多功能流水线中的各段可以按照不同的方式连接，同时执行多种功能

**单功能：**只能完成一种固定功能的流水线

**多功能：**流水线的各段可以进行不同的连接，以实现不同的功能

**10**、锁存技术

？？？PPT找不到……

在流水线各段之间加上锁存器（流水寄存器）

**11**、寄存器换名技术

通过改变指令中操作数的名来消除名相关，对于寄存器操作数进行换名称为寄存器换名，可消除名相关

**12**、定向技术

在某条指令产生计算结果之前，后面等待使用该结果的指令并不一定立即需要该结果，如果能够将该计算结果从其产生的地方（ALU出口）直接送到其他指令需要它的地方（ALU入口），那么就可以避免停顿，减少数据冲突引起的停顿。

**13**、**ROB**技术（**re-order-buffer** ）

用于基于硬件的前瞻执行，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、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（ReOrder Buffer）的缓冲器中。等到相应的指令得到“确认”（commit）（即确实是应该执行的）之后，才将结果写入寄存器或存储器

**14**、分支历史表BHT

最简单的动态分支预测方法，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，有1或者2位的预测位。

**15**、超标量机

在每个时钟周期流出的指令条数不固定，依代码的具体情况而定的处理机。如果流出指令上限为n，就称该处理机为n流出。

**16**、向量机

具有向量数据表示和向量指令的处理机

**17**、时间局部性和空间局部性

时间：程序即将用到的信息很可能就是目前正在使用的信息

空间：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

**18**、数组合并技术、内外循环交换技术、循环融合技术

都是编译器优化

**数组合并：**

/\* 修改前 \*/

int val [ SIZE ];

int key [ SIZE ];

修改后：

struct merge {

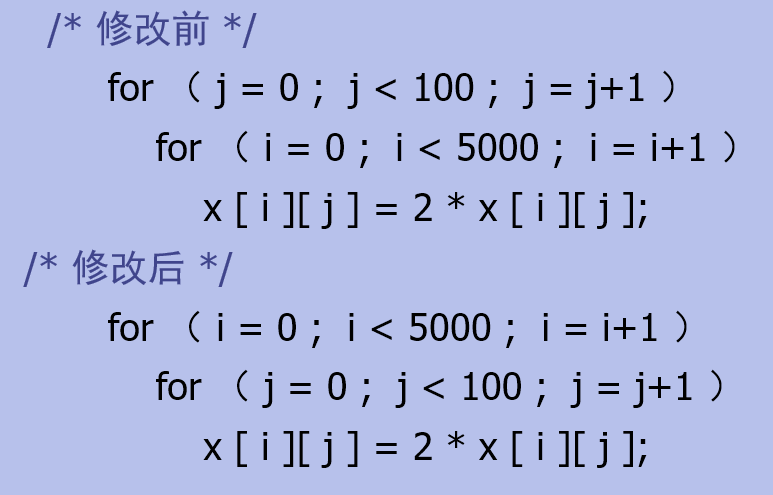
int val ;

int key ;

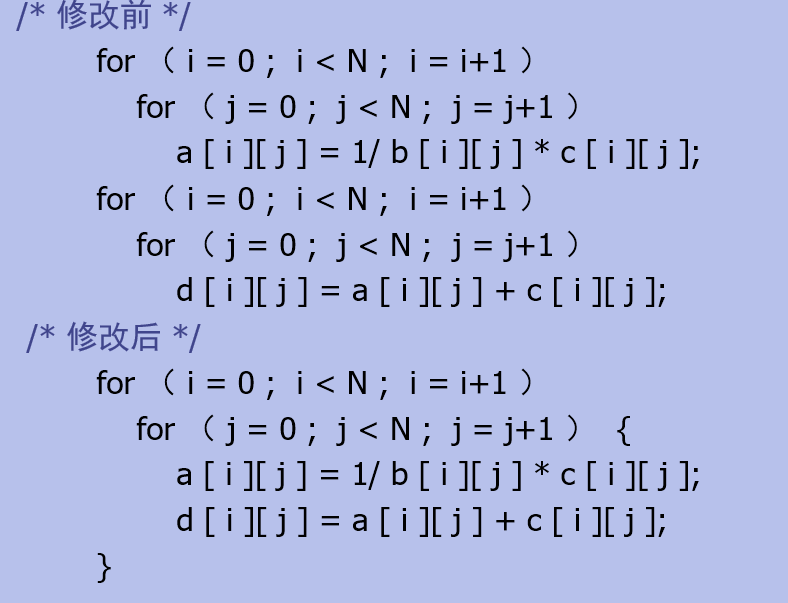
} ;

struct merge merged\_array [ SIZE ];

**内外循环交换技术：**



**循环融合技术：**



**19**、平均访存时间

平均访存时间 ＝ 命中时间＋失效率×失效开销

**20**、强制失效、容量失效、冲突失效

强制：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

容量：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，若又重新被访问，就会发生失效

冲突：在组相联或直接映象Cache中，若太多的块映象到同一组(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

**21**、全相联映像、直接映像、组相联映像

全相联：主存中的任一块可以被放置到Cache中的任意一个位置

直接：主存中的每一块只能被放置到Cache中唯一的一个位置，取模运算（模Cache的块数）

组相联：主存中的每一块可以被放置到Cache中唯一的一个组中的任何一个位置

**22**、非阻塞**CACHE**技术

Cache失效时仍允许CPU进行其他的命中访问。即允许“失效下命中”

**23**、**TLB**

页表缓存、转址旁路缓存，快表，是一种高速缓存，内存管理硬件使用它来改善虚拟地址到物理地址的转换速度。

**23**、平均**CPI**

平均CPI = 执行程序所需的时钟周期数/IC，IC所执行的指令条数

**24**、**SIMD**

在Flynn分类法（按照指令流和数据流的多倍性进行分类）中的单指令流多数据流

**25**、乱序发射/乱序执行/顺序完成

Emmm找不到原话

乱序发射：指令的流出是乱序的

乱序执行：流水线输出端任务流出的顺序与输入端任务流入的顺序可以不同

顺序完成：流水线输出端任务流出的顺序与输入端任务流入的顺序相同

26、指令动态调度

在程序的执行过程中，依靠专门硬件对代码进行调度，减少数据相关导致的停顿，能够处理一些在编译时情况不明的相关（比如涉及到存储器访问的相关），并简化了编译器，使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。但增加了硬件复杂性

27、LRU算法

选择近期最少被访问的块作为被替换的块，选择最久没有被访问过的块作为被替换的块，失效率低

二、简答题（4题，每题8分）

1、对流水线的冲突处理通常有两种方式：静态调度方式和动态调度方式；说明这两种方式，并分别举出其实现方式。

* 静态调度

**依靠编译器**对代码进行静态调度，以减少相关和冲突。它不是在程序执行的过程中、而是在**编译期间进行代码调度和优化**。通过把相关的指令拉开距离来减少可能产生的停顿。

冲突向量法

* 动态调度

在程序的执行过程中，**依靠专门硬件**对代码进行调度，减少数据相关导致的停顿，可以使用记分牌算法和Tomasulo算法。

2、tomasulo采取了什么方法避免三种数据冲突。

记录和检测指令相关，操作数一旦就绪就立即执行，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

3、给出一段有相关性的指令，分析相关性、请重新设计指令顺序（编译器方式），消除相关性

。。。

**4**、计算机系统结构、计算机组成和计算机实现的概念与关系。

**计算机系统结构：**是其所处环境中最高层次的概念，是对计算机系统中各个机器之间界面的划分和定义，以及对各级界面上，下的功能进行分配。

**计算机组成：**是计算机系统结构的逻辑实现，包括机器级内的数据流和控制流的组成以及逻辑设计等。

**计算机实现：**计算机组成的物理实现

**关系：**具有相同系统结构的计算机可以采用不同的组成，一种计算机组成可以采用多种不同的计算机实现。采用不同的系统结构会使可以采用的组成技术产生差异，计算机组成也会影响系统结构计算机组成的设计，其上决定了计算机系统结构，其下又受限于所用的实现技术，他的发展促进了实现技术的发展，也促进了结构的发展。计算机实现，特别是器件技术的发展是计算机系统结构和组成的基础，促进了组成和结构的发展。

**5**、论述**RISC**与的**CISC**技术；讨论**RISC**从哪些方面提高了指令的执行效率，并举例说明。

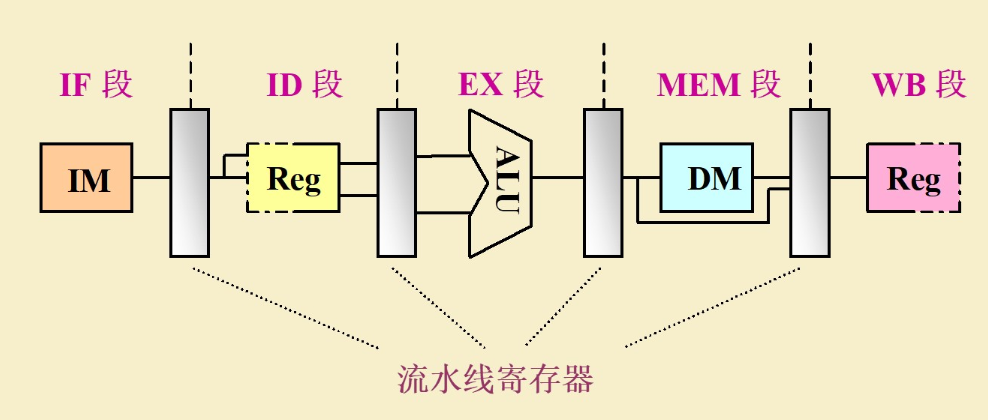
RISC：精简指令集计算机，尽可能地把指令集简化，不仅指令的条数少，而且指令的功能也比较简单

CISC：复杂指令集计算机，增强指令功能，把越来越多的功能交由硬件来实现，并且指令的数量也是越来越多

RISC通过指令条数少而简单，采用简单而又统一的指令格式，并减少寻址方式，指令的执行在单个机器周期内完成，只有load和store指令才能访问存储器，其他指令的操作都是在寄存器之间进行，大多数指令都采用硬连逻辑来实现，来提执行效率。

1981年 ，Berkeley分校的Patterson 等人的32位微处理器RISC I，采用load-store结构，指令字长为32位，采用高效的流水技术。

**6**、简要画出**DLX**多周期（经典**5**段流水）流水线的数据通路图；说明**load**和**store**在每个周期的表现。



**load:**IF取指令，ID译码，EX阶段ALU形成有效地址，DM，访问存储器，WB阶段写回

**store:**IF取指令，ID译码，EX形成有效地址，DM访问存储器，数值写入存储器

**7**、流水线冲突有哪三种？请简述每种流水线冲突。

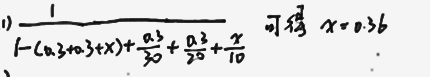
**结构冲突：**因硬件资源满足不了指令重叠执行的要求而发生的冲突

**数据冲突：**当指令在流水线中重叠执行时，因需要用到前面指令的执行结果而发生的冲突

**控制冲突：**流水线遇到分支指令和其他会改变PC值的指令所引起的冲突

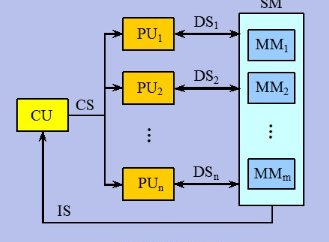
**8**、如果某计算机系统有**3** 个部件可以同时改进，则这**3** 个部件经改进后达到的加速比分别为：***S*1=30**,***S*2=20**,***S*3=10**。

如果部件**1** 和部件**2** 改进前的执行时间占整个系统执行时间的比例都为**30%**，那么，部件**3** 改进前的执行时间占整个系统执行时间的比例为多少，才能使**3** 个部件都改进后的整个系统的加速比***Sn*** 达到**10**？



**9**、**GPU**采用了哪种处理器设计方式作为原型，请简述并画出这种处理器的体系结构原理图。

SIMD处理器设计方式作为原型，单指令流多数据流



符号表示如下：

IS：指令流，DS：数据流，CS：控制流，CU：控制部件，PU：处理部件，MM和SM：存储器

**10**、名相关和数据相关会产生写读冲突、读写冲突、写写冲突。简述这三种冲突，并举例说明是如何造成的。

**前提条件：对于两个指令i，j且i在j前**，有以下三种类型：

写后读冲突RAW：在j写之前j去读，对于真数据相关

写后写冲突WAW：在i写入之前j先写，对应输出相关

读后写冲突WAR：在i读之前j先写，对应反相关

**11**、层次化存储系统存在的理论依据是什么？简要阐述这个依据中的原理。

对存储的容量、速度和价格，三者是矛盾的。采用多种存储器技术，构成所谓的存储层次，可以减缓这个矛盾，“Cache－主存”层次可以弥补主存速度的不足，“主存－辅存”层次可以弥补主存容量的不足。

**12**、写出平均访存时间的公式，从公式的三个变量出发，分别举出一个优化（减少）平均访存时间的技术方案。

**平均访存时间 ＝ 命中时间＋失效率×失效开销**

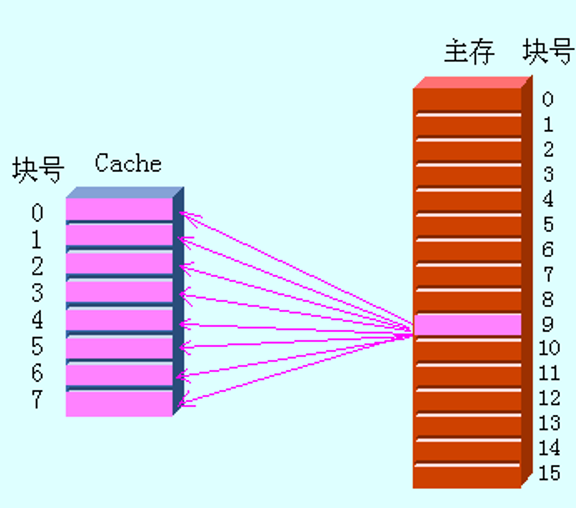
命中时间：虚拟Cache

降低失效率：伪相联Cache

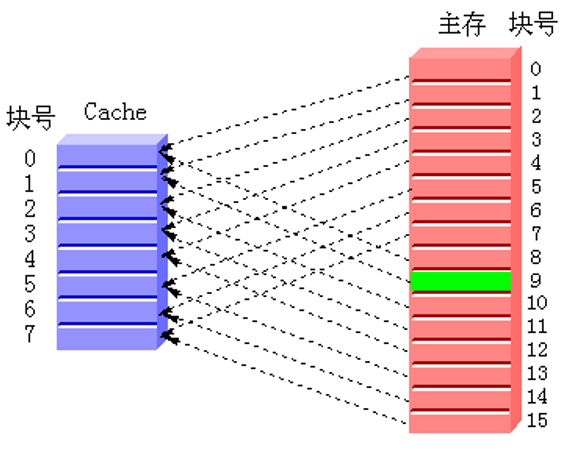
降低失效开销：写缓冲合并

**13**、**CACHE**的地址映像规则有三种：全相联、直接映像与组相联。阐述这三种规则，并用图示法说明三种规则的优缺点。

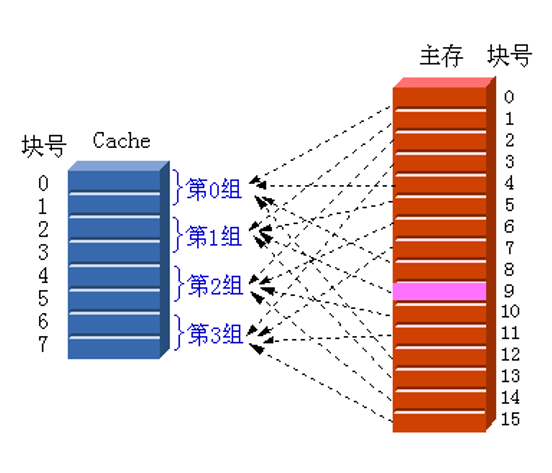
全相联：主存中的任一块可以被放置到Cache中的任意一个位置



直接映像：主存中的每一块只能被放置到Cache中唯一的一个位置



组相联：主存中的每一块可以被放置到Cache中唯一的一个组中的任何一个位置



**14**、有一个Cache存储器，主存有8块(0-7)，Cache有4块(0-3)，采用组相联映像，组内块数为2块，每块大小为16个字节。某程序运行时，要访存主存地址（二进制）为0110110的字节，则访问Cache的哪一块？

16字节所以块内偏移量为4位，组数为2，所以索引占1位，此主存地址索引位为1所以，访问Cache的第1组，即2，3号Cache块，具体哪一块需要看替换算法以及现场。

1. 解决流水线瓶颈问题有哪些方法，应用场合和效果有何异同？

细分瓶颈段和重复设置瓶颈段，细分瓶颈段需要将原有段细分成几个小功能，重复设置瓶颈段，需要重复资源，会增加硬件发复杂度。

1. 简述通过软件（编译器）来减少分支延迟的3种静态方法及它们的共同特点。

**预测分支失败：**允许分支指令后的指令继续在流水线中流动，就好象什么都没发生似的若确定分支失败，将分支指令看作是一条普通指令，流水线正常流动

**预测分支成功：**假设分支转移成功，并从分支目标地址处取指令执行。

起作用的前题：先知道分支目标地址，后知道分支是否成功

**延迟分支：**从逻辑上“延长”分支指令的执行时间。把延迟分支看成是由原来的分支指令和若干个延迟槽构成，不管分支是否成功，都要按顺序执行延迟槽中的指令

**共同点：**

* 对分支的处理方法在程序的执行过程中始终是不变的，是静态的
* 要么总是预测分支成功，要么总是预测分支失败

17、在降低Cache失效率的方法中，对于给定的Cache容量，当块大小增加时，失效率开始是下降，后来反而上升了。解释Cache失效率为什么出现这样的变化?

一方面它减少了强制性失效；另一方面，由于增加块大小会减少Cache中块的数目，所以有可能会增加冲突失效

18、简要说明提高计算机系统并行性的3 种技术途径，并各举一例：

* 时间重叠
* 资源重复
* 资源共享

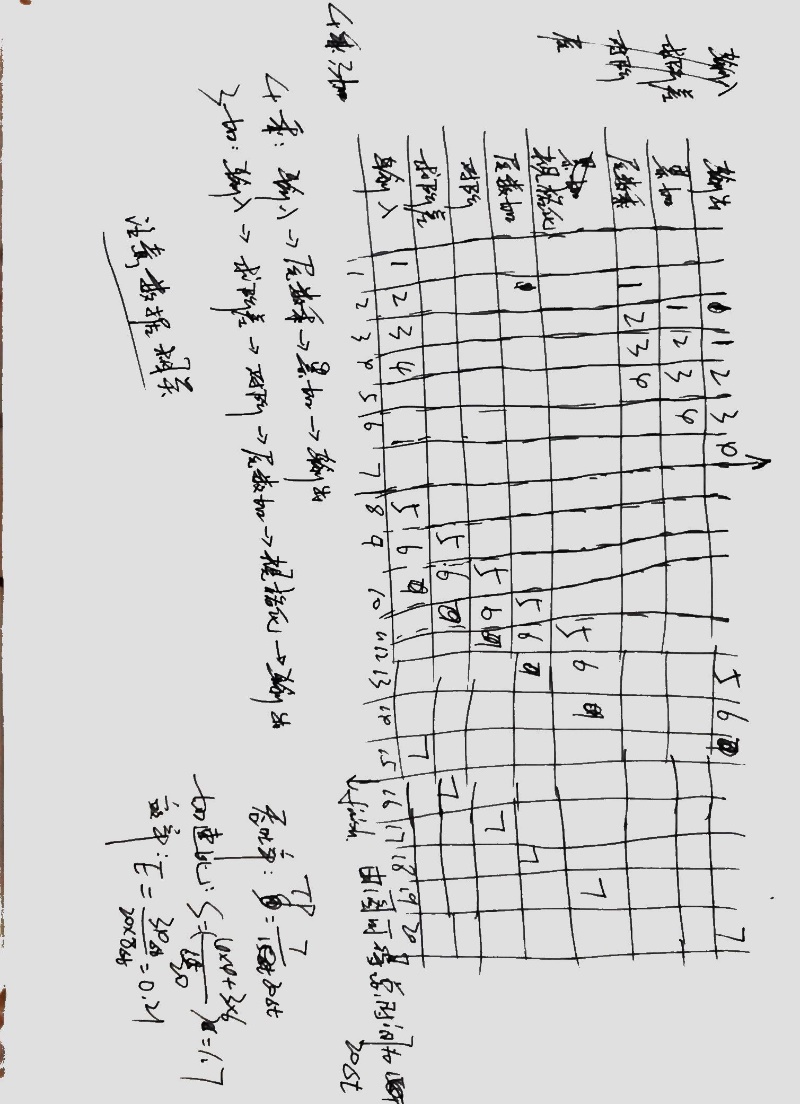
三综合题（**4**题 **38**分，）

**1**、计算机运行以下指令：

**线性多功能静态流水线**，输入任务是不连续的情况，画出该计算的时空图,并计算流水线的吞吐率、加速比和效率。 用TI－ASC计算机的多功能静态流水线计算两个向量的点积：Z＝AB＋CD＋EF＋GH



为方便流水线执行，采用先计算AB，CD，EF，GH,计算AB+CD，然后计算EF+GH，最后计算两者和，4个乘运算，3个加运算



**2**、一条有**4**个流水段的非线性流水线，每一段的延迟时间相等，预约表如下：



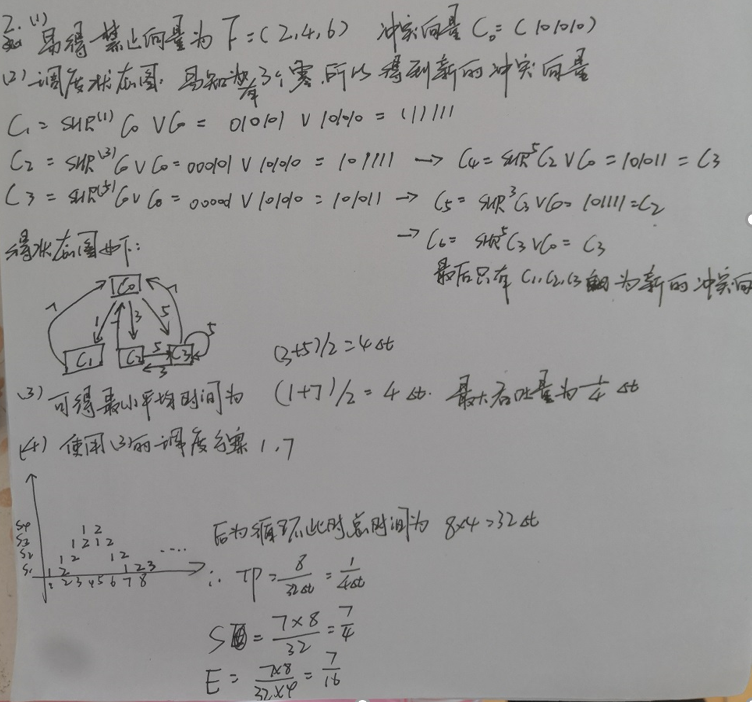
（**1**）写出禁止向量和冲突向量

（**2**）画出调度状态图

（**3**）求出最大吞吐量

（**4**）按最优调度连续输入**8**个任务，实际吞吐量

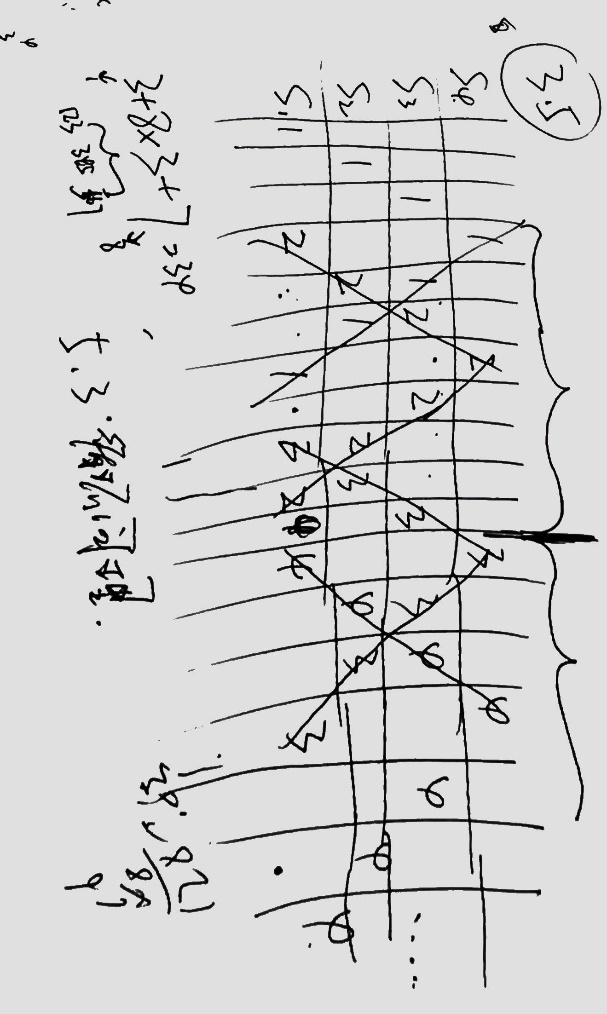
加速比和效率各为多少



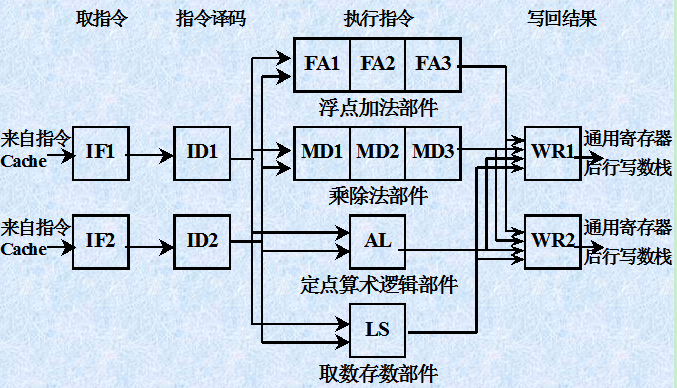
[**https://www.nowcoder.com/questionTerminal/8ca357e9514846849131528ec93bbfbd**](https://www.nowcoder.com/questionTerminal/8ca357e9514846849131528ec93bbfbd)

**有两条路，4问两个都要算一下，上面只有1，7的**

**3，5如下：**



**3**、超标量机的相关性问题以及调度



计算机运行以下指令：

I1：LOAD R1, A ；R1←(A)

I2：FADD R2, R1 ；R2←(R2)＋(R1)

I3：FMUL R3, R4 ；R3←(R3)×(R4)

I4：FADD R4, R5 ；R4←(R4)＋(R5)

I5：DEC R6 ；R6←(R6)－1

I6：FMUL R6, R7 ；R6←(R6)×(R7)

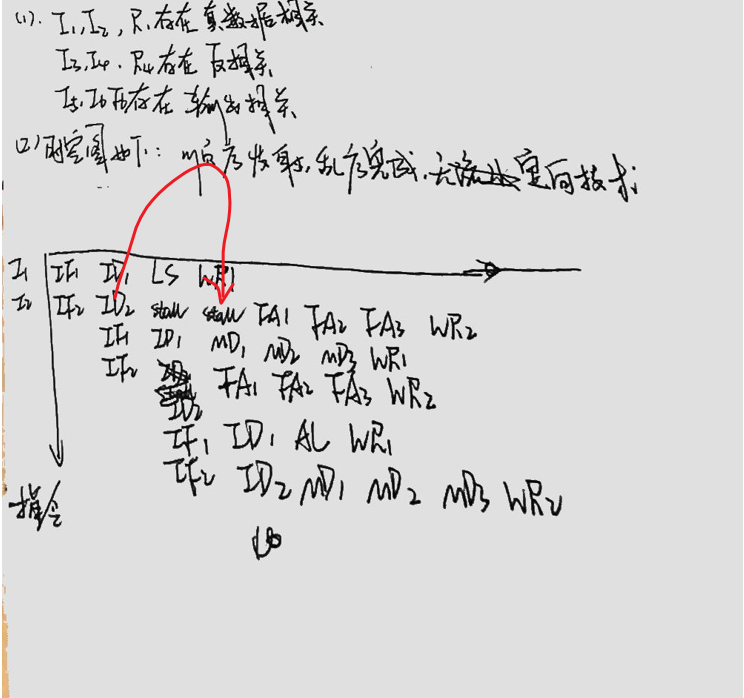
（1）请列出程序代码中可能出现的数据相关及相关类型。

（2）当程序通过下图的双发射超标量机时，请采用顺序发射乱序完成的方式画出指令流水时空图。

(流水线没有使用定向技术。)

**有点迷**

[**https://www.nowcoder.com/questionTerminal/1213c6bc71db437987549a69949460f1**](https://www.nowcoder.com/questionTerminal/1213c6bc71db437987549a69949460f1)



**4**、**CACHE**映像算法

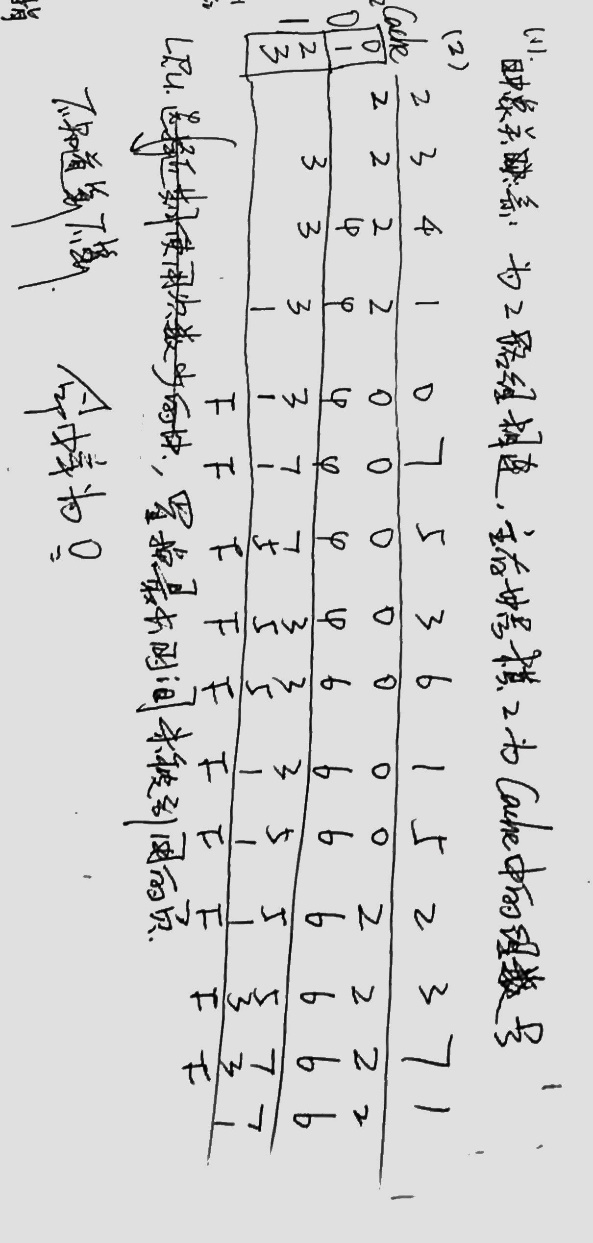
有一个Cache存储器，主存有8块(0-7)，Cache有4块(0-3)，采用组相联映像，组内块数为2块。采用LRU（近期最久未使用）替换算法。（12分，（1）题4分，（2）题8分）

(1)指出主存各块与Cache各块之间的映像关系。

(2)某程序运行过程中，访存的主存块地址流为：

2， 3， 4， 1， 0， 7， 5， 3， 6， 1， 5， 2， 3， 7， 1

说明该程序访存对Cache的块位置的使用情况，指出发生块失效且块争用的时刻，计算Cache命中率。



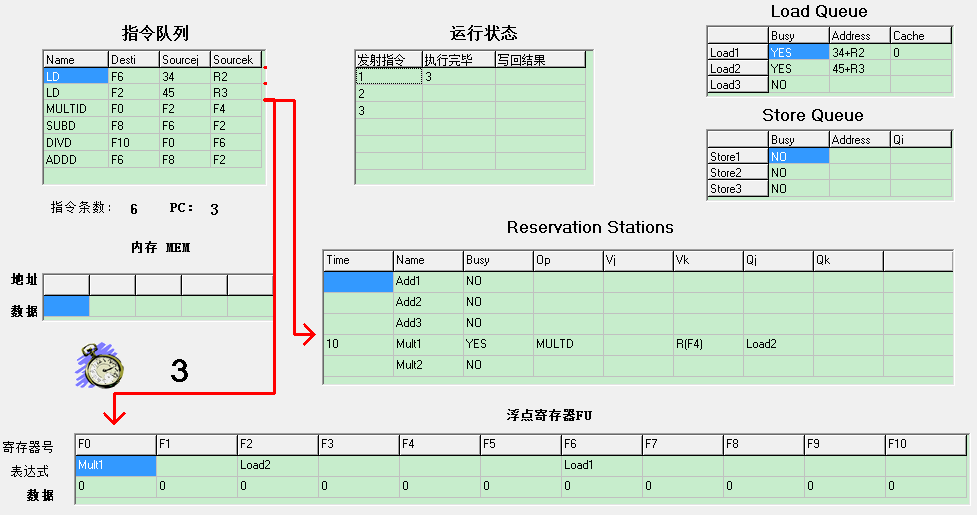
1. 举三个例子说明系统中采用软件来提高性能的方法和效果。
   1. 降低Cache失效率采用的编译器优化
   2. 降低Cache失效率采用的编译器控制的预取
   3. 动态分支预测技术中的BHT
2. 举三个例子说明系统中采用硬件来提高性能的方法和效果。
3. 降低Cache失效率采用的硬件预取
4. 指令级并行的Tomasulo算法
5. 基于硬件的前瞻执行

**7、**tomasulo算法的第3个时钟周期的指令状态，保留站状态，和寄存器结果状态如下图所示；

（其中Op表示现在保留站中正在工作的指令,Vj，Vk表示已经准备好的操作数，Qj,Qk表示已发射但未准备好的操作数）。已知load 执行延时2个cycles，add（sub）执行延时2个cycles，mul 执行延时10个cycles，div 执行延时40个cycles。

要求：

1. 写出tomasulo算法的核心思想。（4分）
2. 写出第4个时钟周期的指令运行状态，保留站状态，和寄存器结果状态，并说明原因。（6分）

****

8、某台主频为400MHz的计算机执行标准测试程序，程序中指令类型、执行数量和平均时钟周期数如下：

|  |  |  |
| --- | --- | --- |
| 指令类型 | 指令执行数量 | 平均时钟周期数 |
| 整数 | 45000 | 1 |
| 数据传送 | 75000 | 2 |
| 浮点 | 8000 | 5 |
| 分支 | 2000 | 2 |

求该计算机的平均CPI、MIPS和程序执行时间(单位：us)。

指令总数为45000+75000+8000+2000=130000

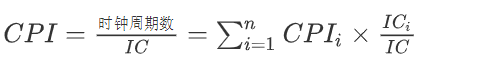
CPI = 1\*(45000/130000)+2\*(75000/130000)+5\*(8000/130000)+2\*(2000/130000)= 1.838

MIPS = 400MHZ/(1.838\*10^6)=

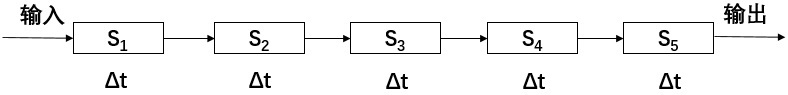
时间T = IC \* CPI \* 1/400MHZ =

9、一台1600MHz计算机执行测试程序，指令类型、执行数量和平均时钟周期数如下



已知平均CPI等于2，则浮点指令的CPI为( )  


10、



一条各流水段执行时间均等的5段线性流水线，各段的执行时间均为Δt，其连续执行了10个任务，且不考虑数据与控制冲突，则其实际加速比为 。（精确到小数点后2位）

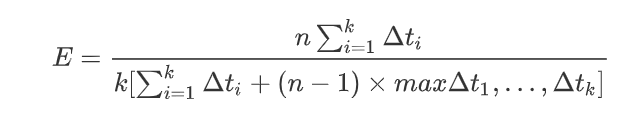
各段时间相等的加速比：



S = (10\*5)/(5+10-1) = 3.57

11、

一条各流水段执行时间不完全相等的5段线性流水线，假设其第1、3、4、5段的执行时间为Δt，第2段的执行时间为2Δt，其连续执行了5个任务，且不考虑数据与控制冲突，则其实际效率为\_\_\_。（精确到小数点后2位）



E = (5\*6)/(5\*(6+4\*2)) = 3/7

12、假设分支目标缓冲的命中率为95%，程序中无条件转移指令的比例为5%，没有无条件转移指令的程序的CPI值为1。假设分支目标缓冲中包含分支目标指令，允许无条件转移指令进入分支目标缓冲，则程序的CPI值为?假设原来的CPI=1.2。（精确到小数点后2位）



设每条无条件转移指令的延迟为x，有：

1+x\*5% = 1.2 => x = 4

所以缓冲包含目标指令的CPI为：（命中时的转移指令延迟为0）

1 + 0.05\*0.95\*0 + 0.05\*0.05\*4 = 1.01

13、假设有一条长流水线，仅仅对条件转移指令使用分支目标缓冲。假设分支预测错误的开销为4个时钟周期，缓冲不命中的开销为3个时钟周期。假设命中率为95%，预测精度为95%，分支频率为10%，没有分支的基本CPI为1。程序执行的CPI为（精确到小数点后3位）



CPI = 1 + 0.1\*(0.95\*0.05\*4+0.05\*3)=1.034

14、考虑某两级cache，第一级为L1，第二级为L2，两级cache的全局不命中率分别是5%和2%，假设L2的命中时间是10个时钟周期，L2的不命中开销是200时钟周期，L1的命中时间是1个时钟周期，平均每条指令访存1.4次。问：每条指令的平均停顿时间是个时钟周期？

存储器停顿时钟周期数＝访存次数×失效率×失效开销

1.4\*(0.05\*10+0.02\*200)=6.3