

# 往年题整理

## 23年

### 一、简答题

- 1.简述计算机体系结构层次
- 2.根据CPU性能公式，描述CISC和RISC在计算机性能上的特点
- 3.分析Cache块不断增大后，失效率先减后增的原因
- 4.描述通道的工作流程

### 二、计算题

1.给一块晶片板（面积16），上面放且只放一个大晶片（面积 $n^2$ ， $n=1, 2, 3$ ）和若干小晶片（面积为1），大晶片可以处理串行指令（占比10%），小晶片可以处理并行指令（占比90%），任何指令执行的速度与相应晶片的面积的平方根成正比，且除了执行指令时间外无任何额外时间。

- （1）求怎么设计，可以使得晶片板加速比最高
- （2）假如全放小晶片，其中一个处理串行指令，剩下的处理并行指令，问加速比多少，和第一问的设计哪个更好
- （3）假如串行指令占比4%，并行指令占比96%，问怎么设计加速比最高，这三问中那个设计最好

2.通道计算。现有一通道， $T_s=2\mu s$ ， $T_d=2\mu s$ ，给出了若干设备（数据传输速率（KBps）从100~10不等），问：

- （1）采用字节多路通道，最少选四个设备，怎么选才能使得总流量最高
- （2）采用数组多路通道， $k=521B$ ，问哪些设备可以被使用

3.一静态流水线，加法使用1345段，乘法使用125段，第三段用 $2\Delta t$ ，其余段用 $1\Delta t$ ，现有一计算 $(a_i+b_i)$ ，问：

- （1）画出流水线时空图，求出吞吐率、加速比、效率
- （2）假如使用动态流水线，求出吞吐量、加速比、效率

4.一组向量指令，和课本的比较像，访存用6个时间单位，加法用6个，乘法用7个。分别求出使用向量链接技术和不使用所用的时间

5.一条由 4 个功能段组成的非线性流水线的预约表如下，每个功能段的延迟时间都为  $\Delta t$ ，

时间	1	2	3	4	5	6	7
s1	x						x
s2		x			x		
s3			x	x			
s4					x	x	

- (1) 写出流水线的禁止向量和初始冲突向量
- (2) 画出调度流水线的状态图
- (3) 求流水线的最小启动循环
- (4) 求最小的恒定循环

6. 互联函数，共32个处理机（标号0~31），求： $C_2(12)$ 、 $\beta_3(9)$ 、 $\sigma^3(8)$ 、 $PM2I_{+2}(28)$ 、 $C_1(\sigma(4))$

## 21年

### 一、简答题

（这部分和往年题一模一样）

1. 计算机系统层次结构
2. 记不清了（可以查一下去年/前年的题 考点完全一致）
3. 通道的工作方式
4. 流水线的瓶颈是什么？怎么消除？

二、求CPI 比较不同指令比例下，两种架构的执行速度

三、向量链接技术：求利用向量链接技术和不用向量链接技术时，分别的总执行时间（向量长度要超过向量寄存器）

四、TLB+LRU，给定一个读页的顺序，计算命中率、什么时候命中、什么时候缺页、第二问是主存和TLB最终状态等

五、禁止表、冲突向量、状态图、最小启动循环及其启动距离、最小恒定启动循环、在两种调度方式下，执行10个任务总执行时间

## 19年

### 一、概念题

1. 计算机系统结构中的多级层次结构
2. 流水线的瓶颈是什么，以及改进的方法
3. 两个局部性原理以及在虚拟存储系统中的应用
4. 通道处理机传输数据的主要过程

## 二、计算题

amdahl定律应用

## 三、计算题

五个中断向量，中断响应顺序从高到低为 1 2 3 4 5

1、更改屏蔽字，是优先级为 x x x x x

2、使用（1）中的优先级，程序运行时 2 4同时请求，在处理中断2时，1 2 5同时请求，画图

3、使用改变处理机优先级的方法来处理中断

## 四、计算题

非线性流水线预约表

1、禁止向量，冲突向量，状态图

2、求最短平均启动距离

3、利用最短启动平均距离，运行10个程序，求吞吐率、加速比、效率

## 五、计算题

向量处理机性能评价：长度为200的向量，在采用连接技术和不采用链接技术两种情况下的运行时间。

# 18年

第一题考amdahl定律计算题，属于简单题。其中有问到CPI和MIPS。

第二题 考到了循环展开的效率（MD这个知识点完全没有复习到。。。）

第三题 统一cache和指令/数据cache，两种类型的cache分别计算缺失率和缺失代价，这里两种CACHE都分了L1和L2

第四题 GPU,操作系统，CPU都有线程的概念，说一说其中的异同。

第五题 已想不起来。。。。

第六题 问在多核CPU中cache一致性是如何实现的。

第七题 Tomasulo算法，给一小段代码以及执行情况，填写保留站的状态表。