计算机组成与原理

1.流水线的计算：（默认使用理论公式，无答案的时候考虑实践公式）

理论公式：1 条指令执行时间+（指令条数-1）×流水线周期

实践公式：指令段数×流水线周期+（指令条数-1）×流水线周期

流水线建立时间=第一条指令执行时间

流水线周期时间=最长的一段时间

2.流水线吞吐率：TP=指令条数/流水线执行时间（使用的是理论公式）

3.最大吞吐率计算：$TPmax = 1/t$ =流水线周期的倒数（t 为流水线周期）

4.流水线加速比的基本公式：S=不使用流水线执行时间/使用流水线执行时间

例：某计算机系统采用 4 级流水线结构执行命令，设每条指令的执行由取指令（2Δt）、分析指令（1Δt）、取操作数（3Δt）、运算并保存结果（2Δt）组成（注：括号中是指令执行周期）。并分别用 4 个子部件完成，该流水线的最大吞吐率为（ ）；若连续向流水线输入 5 条指令，则该流水线的加速比为（ ）。

解析：流水线的计算公式是：1 条指令执行时间 + (指令条数-1)*流水线周期

首先是第 1 条指令的执行时间，也就是把 4 段的时间加起来（2+1+3+2）Δt

流水线周期时间=最长的一段时间 =3Δt

流水线时间=（2+1+3+2）Δt + (n-1) ×3Δt

吞吐率=指令条数/流水线执行时间=$\frac{n}{(2+1+3+2)\ \Delta t + (n-1)\times 3\Delta t}=\frac{n}{5\Delta t+3n\Delta t}=\frac{1}{3}\Delta t$

加速比=不使用流水线执行时间/使用流水线执行时间=$\frac{(2+1+3+2)\ \Delta t\times 5}{(2+1+3+2)\ \Delta t + (5-1)\times 3\Delta t}$=2:1

5.单缓冲区计算公式：

单缓冲区：输入时间 T 传送时间 M 处理时间 C

传送时间 M 和输入时间 T 的和与处理时间 C 两者之间构成流水线

计算公式：（M+T+C）+（n-1）*流水线周期

n 为指令条数，对比 M+T 和处理时间 C 值，谁长谁为流水线周期。一般都是 M+T 为流水线周期

6.双缓冲区计算公式：

双缓冲区：输入时间 T 传送时间 M 处理时间 C

传送时间 M、输入时间 T 和处理时间 C 三者之间构成流水线

计算公式：（M+T+C）+（n-1）*流水线周期

n 为指令条数，对比输入时间 T、传送时间 M、处理时间 C 三者的值 ，谁长谁为流水线周期。

例：假设磁盘块与缓冲区大小相同，每个盘块读入缓冲区的时间为 12us，由缓冲区送至用户区的时间是 8us，在用户区内系统对每块数据的处理时间为 2us。若用户需要将大小为 20 个磁盘块的 abc1 文件逐块从磁盘读入缓冲区，并送至用户区进行处理，那么采用单缓冲区需要花费的时间为（ ）us；采用双缓冲区需要花费的时间为（ ）us。

解析：当采用单缓冲区时，由于将盘块读入缓冲区与将数据从缓冲区转到用户区，都要用到同一个缓冲区，所以只能把这两步作为流水线的一个段。所以计算方式为：（12+8+2）+（20-1）*（12+8）=402

当采用双缓冲区时，读入缓冲区与将数据从缓冲区转到用户区可以作为流水线的两个段，所以计算方式为：（12+8+2）+（20-1）*12=250

7. 如果以 h 代表对 Cache 的访问命中率，t1 表示 Cache 的周期时间，t2 表示主存储器周期时间，以读操作为例，使用“Cache+主存储器”的系统的平均周期为 t3，则：t3=h×t1+(1-h)×t2

其中，(1-h）又称为失效率（未命中率)。

例：设某计算机主存的读/写时间为 500ns，有一个指令和数据合一的 Cache，已知该 Cache 的读/写时间为 20ns，取指令的命中率为 99%，取数的命中率为 90%。在执行某类程序时，约有 1/8 指令需要额外存/取一个操作数。假设指令流水线在任何时候都不阻塞，则设置 Cache 后，每条指令的平均读取约为（ ）ns。

解析：（500×1%+20×99%）+（500×10%+20×90%）×1/8=（5+19.8）+（50+18）/8=24.8+8.5=33.3

8.海明码：$2^r \geqslant m+r+1$，其中 r 为校验位，m 为信息位

例：海明码利用奇偶性检错和纠错，通过在 n 个数据位之间插入 k 个检验位，扩大数据编码的码距。若 n=48，则 k 应为（ ）。

解析：此题中 n = 48，校验位个数为 k，则 n+k+1≤2k，即 48+k+1≤2k，则 k 为 6。

9.CRC 冗余循环：

CRC 的编码方法是：在 k 位信息码之后拼接 r 位校验码。应用 CRC 码的关键是如何从 k 位信息位简便地得到 r 位校验位（编码），以及如何从 k+r 位信息码判断是否出错。

循环冗余校验码编码规律如下：

①把待编码的 N 位有效信息表示为多项式 M（X）；

②把 M（X）左移 K 位，得到 M（X）×X^K，这样空出了 K 位，以便拼装 K 位余数（即校验位）；

③选取一个 K+1 位的产生多项式 G（X），对 M（X）×X^K做模 2 除；

④把左移 K 位以后的有效信息与余数 R（X）做模 2 加减，拼接为 CRC 码，此时的 CRC 码共有 N+K 位。

例：循环冗余校验码（Cyclic Redundancy Check，CRC）是数据通信领域中最常用的一种差错校验码，该校验方法中，使用多项式除法（模 2 除法）运算后的余数为校验字段。若数据信息为 n 位，则将其左移 k 位后，被长度为 k+1 位的生成多项式相除，所得的 k 位余数即构成 k 个校验位，构成 n+k 位编码。若数据信息为 1100，生成多项式为 X3+X+1（即 1011），则 CRC 编码是（ ）。

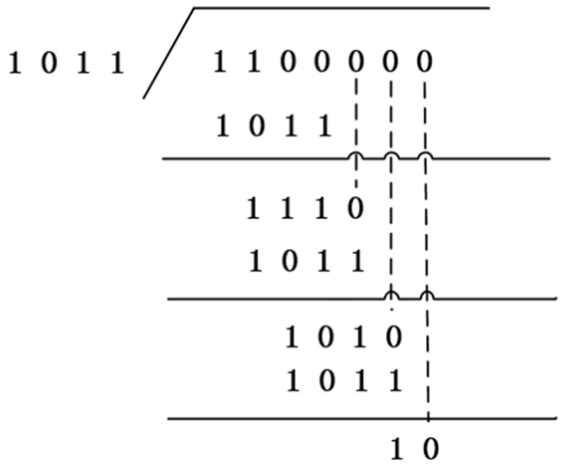
解析：编码流程为：

1、在原始信息位后加 k 个 000，即 1100000。

2、将 1100000 与生成多项式 1011 做模 2 除法，得到余数为 010。

3、将原始信息位与余数连接起来得到：1100010。

生成多项式 x^3+x+1
系数 1011 余数则为3位
已知信息位1100



余数：010
编码：1100010
（编码为原始信息位+余数）

操作系统

1.银行家算法：系统不可能发生死锁的最小资源数：（w-1）× m + 1 ⩽ n

其中 w 为可需要的资源数，m 为进程数，n 为总共资源数

例：某系统中 11 台打印机，N 个进程共享打印机资源，每个进程要求 3 台。当 N 的取值不超过（ ） 时，系统不会发生死锁。

解析：假设有 m 个资源，每个进程最多可申请 k 个资源，则系统要想绝对避免死锁的发生，允许的最多进程数为 1+(m-k)/(k-1)，当后面一项是小数时，总是取为小整数。则有 1+(11-3)/(3-1)=5，因此当 N 的取值不超过 5 时，系统不会发生死锁。

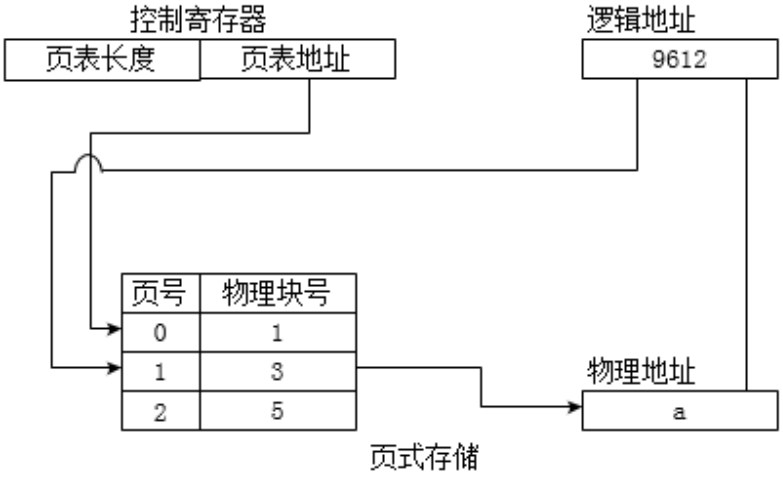
2.逻辑地址转换为物理地址的计算：

逻辑地址=页号+页内地址

物理地址=块号+页内地址

实质：就是将页号换成块号，一般给出逻辑地址和对应的页面大小

例：页式存储系统的逻辑地址是由页号和页内地址两部分组成的，地址变换过程如图所示。假定页面的大小为 8KB，图中所示的十进制逻辑地址 9612 经过地址变换后，形成的物理地址 a 应为十进制（）。



解析：由题目已知页面大小为 8KB，因为 8KB=213，所以页内地址有 13 位。现在把逻辑地址 9612 转成二进制得：10 0101 1000 1100，这里的低 13 位为页内偏移量，最高一位则为页号，所以逻辑地址 9612 的页号为：1 即十进制的 1，所以物理块号为 3，转为二进制得：11。把物理块号和页内偏移地址拼合得：110 0101 1000 1100，转为十进制得：25996。

数据库系统

1.关系模式 R<U，F>来说有以下的推理规则：

A1.自反律(Reflexivity)：若 YSXSU,则 X→Y 成立。

A2.增广律(Augmentation)：若 ZSU 且 X→Y,则 XZ→YZ 成立。

A3.传递律（Transitivity)：若 X→Y 且 Y→Z,则 X→Z 成立。

根据 A1，A2，A3 这三条推理规则可以得到下面三条推理规则：

合并规则：由 X→Y，X→Z，有 X→YZ。（A2，A3）

伪传递规则：由 X→Y，WY→Z，有 XW→Z。（A2，A3）

分解规则：由 X→Y 及 Z 三 Y，有 X→Z。(A1，A3)

计算机网络

1.子网划分的计算：根据划分的子网数 A，推断出 $2^x < A < 2^y$,那么就 需要知道至少要划分 y 位作为划分子网。

2.主机的可用个数计算：可用主机数=2 主机位 -2（减去 2 是因为全 0 和全 1 的主机不

可用）

例：某公司网络的地址是 192.168.192.0/20，要把该网络分成 32 个子网，则对应的子网掩码应该是（ ），每个子网可分配的主机地址数是（ ）。

解析：题目中，原网络地址是 192.168.192.0/20，即 IP 地址前 20 位是网络号。目前要将此网络分成 32 个子网，2^5=32，则需要取 5 位主机号充当网络号，即前 25 位为网络号。此时的子网掩码应是前 25 位为 1，后 7 位为 0 的地址，即：255.255.255.128。

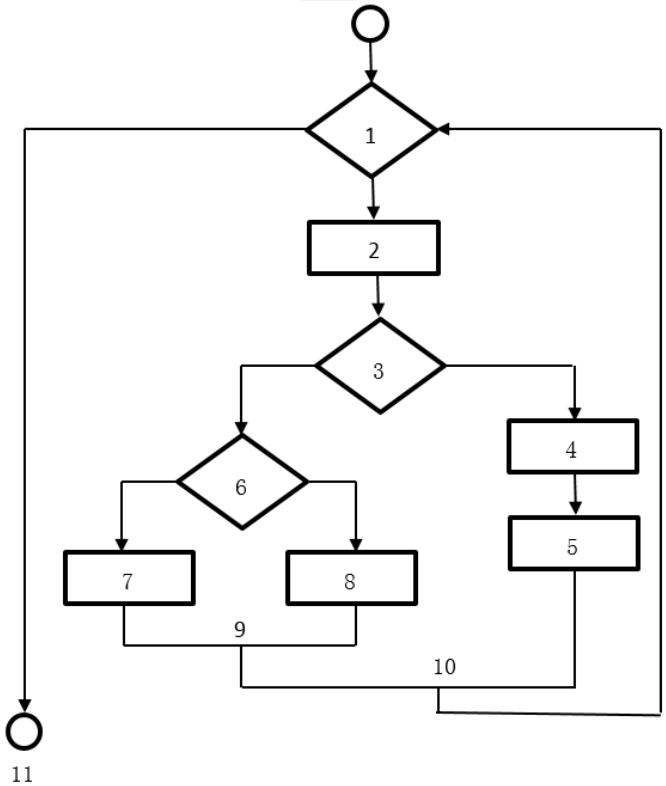
每个子网可分配的主机地址数为：2^7-2=126。

注：每个子网中，主机号为全 0 或全 1 的地址不能使用，所以要减 2。

软件工程

1. Macabe 环路复杂度：V（G）=m+n-2，其中 V（G）是有向图中 G 中的环路个数，其中 m 是 G 有限弧数，n 是 G 中节点数。

例：使用 McCabe 方法可以计算程序流程图的环形复杂度，下图的环形复杂度为（ ）。



解析：方法一：图 G 的环形复杂度 V(G)=E-N+2，其中，E 是流图中边的条数，N 是结点数。边：起点-1、1-2、2-3、3-4、4-5、3-6、6-7、6-8、7-1、8-1、5-1、1-11 。结点：起

点、1、2、3、4、5、6、7、8、11。本题中，E=12，N=10，所以 V（G）=12-10+2=4。方法二： 图 G 的环形复杂度 V(G)=P+1， 其中，P 是流图中判定结点的数目。判定结点：1、3、6。本题中，P=3，所以 V（G）=3+1=4。考试中建议两种方法一起用，相互验证。

项目管理

1.销售额=固定成本+可变成本+税费+利润【正常情况下】

销售额=固定成本+可变成本+税费【盈亏平衡时】

2.三点估算法：最可能的时间=(最乐观时间+4*最可能时间+最悲观时间)/6

例：作业 C 所需的时间，乐观估计为 5 天，最可能为 14 天，保守估计为 17 天。则作业 C 的最可能的工期为（ ）。

解析：使用三点估算法计算出 C 的所需天数：（5+14*4+17）/6=13。

系统可靠性分析与设计

1.可靠性的计算公式： $MTTF/(1+MTTF)$ ，其中 MTTF 为平均无故障时间

2.串联系统：$R1, R2 \cdots Rn$进行串联

可靠性：$R=R1\times R2\times \cdots Rn$

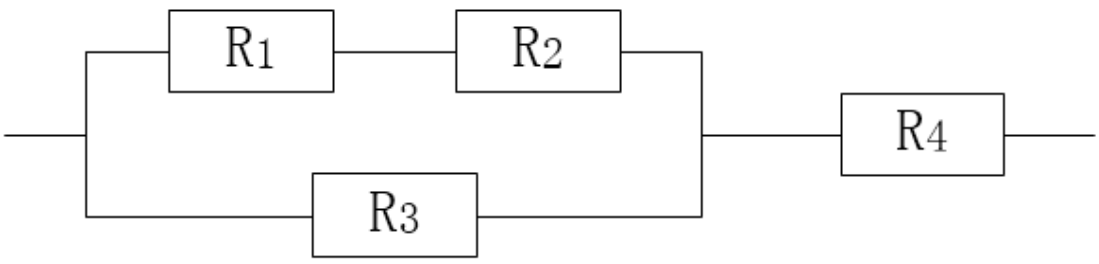
3.并联系统：$R1, R2 \cdots Rn$进行并联

可靠性：$R=1-(1-R1)\times(1-R2)\times\cdots(1-Rn)$

4.混合系统：1 个 R，3 个并联的 R，2 个并联的 R 进行串联

可靠性：$R\times(1-(1-R)^3)\times(1-(1-R)^2)$

例：某计算机系统的可靠性结构如下所示，若所构成系统的每个部件的可靠度分别为 R1、R2、R3 和 R4，则该系统的可靠度为（ ）。



解析：设每个子系统的可靠性分别以 R1 、R2，...，RN 表示，则整个系统用串联方式构造时的可靠度为 R=R1×R2…×…RN，整个系统用并联方式构造时的可靠度为 R=1－（1－R1）（1－R2）...（1-RN）。

题图中，R1，R2 是串联关系，其可靠度为 R1×R2，R3 与 R1、R2 并联后再与 R4 串联，因此整个系统的可靠度为（1－（1－R1R2）（1－R3））R4 。

系统配置与性能评价

1．性能指标计算 MIPS=指令条数/（执行时间× 10^6）=主频/CPI=主频×IPC

例：峰值 MIPS（每秒百万次指令数）用来描述计算机的定点运算速度，通过对计算机指令集中基本指令的执行速度计算得到。假设某计算机中基本指令的执行需要 5 个机器周期，每个机器周期为 3 微秒，则该计算机的定点运算速度为（ ）MIPS。

解析：根据题干描述，假设某计算机中基本指令的执行需要 5 个机器周期，每个机器周期为 3 微秒，则该计算机每完成一个基本指令需要 5*3=15 微秒，根据峰值 MIPS 的定义，其定点运算速度=1 条指令/15 微秒=1/（15*10-6 秒）=1*10-6 百万/（15*10-6 秒）=0.067 百万/秒=0.067MIPS，特别需要注意单位“微秒“和“百万指令数”，在计算过程中恰好抵消。

2．阿姆达尔（Amdahl)解决方案：对系统中某组件采用某种更快的执行方式，所获得的系统性能的改变程度，取决于该组件被使用的频率，或所占总执行时间的比例。加速比计算公式如下：

$$R = \frac{T_p}{T_i} = \frac{1}{(1 - F_e) + F_e / S_e}$$

例：假设某基准程序在一台计算机上的运行时间为 100 秒，其中 80 秒的时间是用来执行乘法操作的，如果希望该程序的速度提高到原来的 4 倍，乘法部件的速度应该是原来的（）倍。

解析：设乘法部件的速度应该是原来的 n 倍，由 Amdahl 定律可知，改进后的基准程序执行时间=80 秒/n+20 秒=100 秒/4 即 80 秒/n+20 秒=25 秒，则 n=16。