

2016-2025 年清华计算机考研 826 真题 (按知识点划分)

数据结构

绪论 (共 14 道题)



判断是否满足 O 记号 (5 道题)

1. 某时间复杂度函数 $f(n)$ 满足 $f(n) = O(g(n))$, 但不一定满足 $f(n) = O(g(n-1))$. () (2017)
2. $n^{\log \log \log n} = O([\log n]!)$ () (2019)
3. $\log^n n = \Theta(n^{\log n})$ () (2020)
4. $[\log n]! = O(n^{\log \log n})$ () (2022)
5. 考查复杂度函数 $f(n) = 2^{\log^2 n}$ 和 $g(n) = n^{2020}$, 从渐进的角度看, 何者增长更快? 为什么? (2021)

主定理 (4 道题)



1. 假设 $T(1) = a$, 无论常数 a 多大, 时间复杂度关系式 $T(n) = T\left(\frac{n}{2}\right) + O(1)$ 的解总是 $O(\log n)$. () (2018)
2. 若某复杂度函数 $f(n)$ 满足: $f(0) = f(1) = O(1)$, 且对于 $n \geq 2$ 满足递推式: $f(n) = 3f\left(\frac{n}{9}\right) + O(1)$, 则就大 O 记号而言, $f(n)$ 属于哪个渐进复杂度层次? 为什么? (2021, 6 分)
3. 若某算法的运行时间满足 $T(n) = 128T\left(\frac{n}{2^{2023}}\right) + {}^{290}\sqrt{n}$, 则 $T(n) = \underline{\hspace{2cm}}$. (2023)
4. $T(1) = O(1)$, $T(n) = 512T\left(\frac{n}{2^{2025}}\right) + {}^{450}\sqrt{(5n-4)(3n+2)}$, 则 $T(n) = O(\underline{\hspace{2cm}})$. (2025)

计算具体算法的时间复杂度 (1 道题)

1. 以下函数时间复杂度为 (2024)

```
void F(int n){//假定字宽足够,运算不会溢出,且每次只需常数时间
    for (int i = 1, j = 1; i < n; j <= 1)
        i <= j;
}
```



就地算法的概念 (1 道题)

1. 就地算法是指空间复杂度为 () (2018)
(A) $O(1)$ (B) $O(n)$ (C) $O(n^2)$ (D) $O(n \log n)$

最长公共子序列 (1 道题)

1. 两个序列的最长公共子序列可能不止一个. () (2025)



Fib(n) 的计算时间 (1 道题)

1. 计算 Fib(n) 至少需要 $\Omega(n)$ 的时间. () (2025)

其他 (1 道题)

1. 若指针 p 指向某一个逻辑地址,那么 $p++$ 就是访问下一个逻辑地址. () (2016)

向量 (共 5 道题)

二分查找 (1 道题)

1. 任何情况下,折半查找都比顺序查找快. () (2018)



Fibonacci 查找 (1 道题)

1. 以前后不同的黄金分割点作为分界点的两个 Fibonacci 查找算法的时间复杂度的常系数相同. () (2018)

对称矩阵的压缩存储 (1 道题)

1. 一个 50×50 的对称矩阵 $a[0,49][0,49]$,每个数占两个地址单元,行优先的前提下压缩成下三角矩阵, $a[0][0]$ 起始下标对应的是十进制的 1000,若 $a[i][j]$ 的地址是 2000 ($i \leq j$),则 $i = \underline{\hspace{1cm}}$, $j = \underline{\hspace{1cm}}$. (2022)

向量的扩容和缩容问题 (1 道题)

1. 向量只要在即将溢出前使容量加倍,并在装填因子不足 50% 时容量减半,便可保证消耗在容量调整的分摊时间复杂度为常数. () (2023)

向量删除时间复杂度 (1 道题)

1. 在长度为 n 的向量中删除第 r 个元素 ($0 \leq r < n$) 需要 $O(1 + r)$ 时间. () (2024)



列表 (共 1 道题)

有序列表的归并 (1 道题)

1. 有序向量的二路归并算法能在线性时间内完成,有序列表则不能. () (2022)

栈和队列 (共 19 道题)

栈混洗, Catalan 数 (8 道题)

1. 五个互异节点构造的二叉搜索树一共有 () 棵. (2017)
(A)20 (B)30 (C)32 (D)42 (E)62
2. 有 2019 个叶节点的真二叉树的数目小于由 2018 对括号可以组成的合法表达式数目. () (2019)
3. 对于叶节点个数为 () 的真二叉树,其数量等于 2018 对括号所组成的合法表达式数量. (2020)

(A)2018 (B)2019 (C)4038 (D)4039

- 由____个元素构成栈混洗序列,与由 2024 个括号构成合法表达式一样多. (2024)
- 由关键码{0,1,2,...,10}组成且所有节点都是偶数度的二叉搜索树共有____棵. (2022)
- {0,1,2,3,4,5,6,7,8}九个关键码构成的真二叉搜索树的种类有()种. (2025)

(A)14 (B)16 (C)512 (D)4862
- <ABCDEFGJ>栈混洗之后不可能出现的结果是() (2025)

(A)[CDBAFGE> (B)[DECFBGA> (C)[EDCFABG> (D)[BDECGFA>
- 入栈顺序为 MAMAMIA,要求出栈顺序也为 MAMAMIA,共有几种方案? (2016)

(A)4 (B)5 (C)6 (D)7



RPN 求值 (6 道题)

- 有如下逆波兰式结果为 2048,?处中的运算符号是多少() (2017) (原题数不对,新威考研复原)

0 ! 2 * 2 2 * 6 + ^ 8 8 ? 32 / *

(A)+ (B)* (C)^ (D)! (E)/
- 若逆波兰式 $0!1+23!4+^*56!7*8!?!/-9+$ 的值等于 2017,则?处的运算符号为() (2018,2020)

(A)加号 (B)减号 (C)乘号 (D)除号 (E)乘方 (F)阶乘
- 如下逆波兰表达式的值若为 2023,则被遮挡的那个运算符是____. (2023,4 分)

0 ! 1 2 + 3 4 + ? + 5 ! 6 7 + 8 - 9 * + -

必须填写运算符及其对应的中文(+加法,-减法,*乘法,/除法,^乘方,! 阶乘)
- 若以下逆波兰表达式值等于 2024,则被遮挡的运算符是() (2024)

$0!1+23!4+^*5!6/-7+8()9+*$

(A)+ (B)- (C)* (D)/ (E)^ (F)!
- RPN“0,!,1,^,2,□,3,!,!,×,4,5,+/,6,×,7,8,+,-,9,×”求值等于 2025,则□中的运算符是() (2025)

(A)+ (B)- (C)* (D)/ (E)^ (F)!

转成 RPN 的意义 (1 道题)

- 相对于常规表达式,逆波兰式在求值计算时为何效率更高? 前者转换为后者所需成本已相当于一
次常规表达式求值,那么转换的意义何在? (2019,4 分)

具体中缀表达式转 RPN (1 道题)

- 考察加(+),减(-),乘(×),除(/),乘方(^),阶乘(!) 运算 (2021,6 分)
试将中缀表达式“(0!+1)×2^(3!+4)-(5!/6+(7-(8-9)))”转换为逆波兰表达式,并按如下格式分
段填写:

0	!	1	+	2	3	!	4	+	^	×	5	!	6	/	7	8	9	-	-	+	-
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

非法表达式求值 (1 道题)

- 非法表达式(12) 3 + ! 4 * + 5,执行 evaluate 算法后的结果是() (2018)

(A)99 (B)89 (C)88 (D)98



栈的 pop 和 push 接口复杂度 (1 道题)

1. 基于列表派生出的栈,无论将哪一端当作栈顶,pop 和 push 接口的时间复杂度都是 $O(1)$. () (2024)



函数调用栈 (1 道题)

1. 程序执行过程中,调用栈中若同时有多帧对应同一函数,则它们必然依次紧邻排列. () (2019)

树和二叉树 (共 14 道题)

二叉树遍历和重构 (6 道题)

1. 某二叉树中序遍历序列为 {D,B,A,E,C,F},层次遍历序列为 {A,B,C,D,E,F},能否唯一确定一棵二叉树? 能则给出步骤,不能则请构造其中一棵. (2016)
2. 二叉树遍历 (题目不完整) (2016)
给了一个递归算法,问访问节点的顺序,树的样子是一棵深度为 4 的二叉树.
3. 对于某二叉树,知道先序遍历序列和后序遍历序列不能确定其层序遍历序列. () (2019)
4. 证明 (5 分) (2020)
给定一棵二叉树的先序遍历序列和后序遍历序列,通过先序遍历序列和后序遍历序列能否确定唯一的层次遍历序列,若可以给出证明,不可以则说明理由.
5. 先序遍历序列 FCDEBGA,后序遍历序列 DCGBAEF,则层次遍历序列为____. (2025)
6. 对于叶节点数量为 2018 的树,层序遍历使用的辅助队列规模绝对不超过 2018. () (2019)

PFC 编码,哈夫曼编码 (6 道题)

1. 哈夫曼树深度更小的节点的权值可能小于深度更大的节点的权值. () (2017)
2. 如果互换 PFC 的不同深度节点,一定会破坏其性质. () (2018)
3. 交换某哈夫曼树不同深度的子树,其编码成本必然增加. () (2019)
4. 9 个字符出现频率为 0,1,1,2,3,5,8,13,21,其哈夫曼编码 () (2020) (频率 0 可能是出题人欠考虑)
(A)最大长度是 6 (B)最大长度是 7 (C)最大长度是 8 (D)最大长度是 9
5. 若仅使用 9 个字符,且他们出现的频率分别为 {1,1,2,3,5,8,13,33,36},则按照 Huffman 算法对长度为 2023 的随机字符串做编码,所需比特位数的期望值是____. (2023)
6. 出现频率为 {1,2,3,3} 的一组字符共有 () 种 Huffman 编码方案. (2024)
(A)16 (B)24 (C)32 (D)40



其他 (2 道题)

1. 将以 x 为根的子树节点个数记为 $x.size$,如果将二叉树中满足 $x.size*2 \leq x.parent.size$ 的非根节点 x 称作“轻节点”,那么在包含 2024 个节点的二叉树中,一个节点至多可能有 () 个轻的真祖先. (2024)
2. 一个二叉树中所有节点都是偶数度,最深的叶节点需要经过 11 条边才能到达树根,则该二叉树最少有____个节点. (2022) (网传回忆版,22 上过考场的坚定地说原题完全不是这个,暂且就按这个算)



搜索树 (共 11 道题)

BST (1 道题)



1. 二叉搜索树中关键码最大的节点 () (2016)
- (A)可能有左孩子,没有右孩子 (B)仅有右孩子,没有左孩子
- (C)既有左孩子,又有右孩子 (D)没有左孩子,也没有右孩子

AVL 的插入和删除 (8 道题)

1. 在 AVL 树中, () 节点操作可能会导致两次旋转调整? (2016)
- (A)插入、删除 (B)仅删除 (C)仅插入 (D)插入、删除都不可能
2. 将关键字 1,2,3,...,2016 依次插入初始为空的 AVL 树中,假设只有一个根节点的二叉树高度为 0,那么最终的 AVL 树的高度是多少? () (2017)
- (A)8 (B)9 (C)10 (D)11 (E)12
3. 对于同一个长度为 n 的序列,分别按照递增的顺序和递减的顺序构造 AVL 树,那么“存在正整数 k ,使得 $n = 2^k - 1$ ”是“两次构造的 AVL 树相同”的 () (2018)
- (A)充分不必要条件 (B)必要不充分条件 (C)充分必要条件 (D)非充分也非必要条件
4. AVL 树在插入一个节点后可能引起 $O(\log n)$ 次局部重构. () (2020)
5. AVL 树 (4+2+8=14 分) (2021)

我们知道,AVL 树在删除一个节点后如果失衡,可以通过适当的调整来恢复平衡.

- (1) 为确定应在何处,以何种方式完成这样的调整我们需要在某相邻的三层中确定三个节点,并按高度递减次序分别命名为 g, p 和 v ,试说明在算法中是如何找到它们的?
- (2) 如果将各种调整的方式统一为“(3+4)-重构”,那么在调用 `connect34(a,b,c,T0,T1,T2,T3)` 时,七个参数共有多少种可能的情况?
- (3) 试按如下表格形式,逐行地具体列出这些情况 (次序无所谓)

可能的情况	a	b	c	T0	T1	T2	T3
#1							
#2							
#3							
#4							

6. 删除 AVL 树的节点 x ,导致失衡,确定最低失衡节点 g ,调整平衡后, g 的高度可能不能保持原高度. () (2022)
7. 按照二叉搜索树规则在 AVL 树中插入一个节点后,若造成失衡了,且在定位祖孙三代节点 g, p, v 后,只需要围绕 g 做一次旋转即可使局部恢复平衡了,则在旋转之前节点 p 的平衡因子不可能是零. () (2023)
8. 在 AVL 树一次删除操作过程中,可能需要旋转 $\Omega(\log n)$ 次. () (2024)



Fib-AVL (2 道题)

1. 一棵二叉搜索树有 8 个叶节点,且内部节点平衡因子都是+1,则共有____个节点. (2024)
2. 存在由 () 个节点组成的 AVL,所有内部节点平衡因子都不等于零. (2025)
- (A)32 (B)33 (C)34 (D)35

高级搜索树 (18 道题)

伸展树分摊时间复杂度 (4 道题)

1. 理想随机的伸展树的插入操作的分摊时间复杂度 $O(\log n)$. () (2018)
2. 如果访问的顺序不具备局部性, 则伸展树查找不能保证分摊 $O(\log n)$ 的时间. () (2019, 2023, 2025)

伸展树势能和高度变化问题 (4 道题)



1. 伸展树总体势能不可能超过 $O(n \log n)$. () (2022)
2. 伸展树为单链时, 势能达最小. () (2024)
3. 伸展树势能定义为 $\Phi(S) = \log(\prod_{v \in S} \text{size}(v))$, 对于根结点为 1, 共 2025 个节点的右侧单链形式的伸展树, 查找 2025, 查找之前的势能为 P, 查找之后的势能为 Q, 则 $P - Q = \underline{\hspace{2cm}}$ (写出表达式即可) (2025)
4. 一棵伸展树的某节点经过访问后, 节点被转换到根部, 树的高度不一定降低. () (2022)

B 树 (4 道题)

1. 在有 2016 个关键码的 7 阶 B 树中搜索, 假设根节点常驻内存, 则最多共需启动 () 次 I/O. (2017)
(A)4 (B)5 (C)6 (D)7 (E)8
2. 一个具有 2017 个关键码的 7 阶 B 树, 根节点常驻内存, 则一次查找最多进行 () 次 I/O 操作. (2018)
(A)7 (B)6 (C)5 (D)4
3. 在有 400 个关键码的 20 阶 B 树中进行一次查找, 则最多需要和 个关键码进行比较. (2022)
4. 在一棵 B 树中, 分支最多节点的分支数目不会超过分支最少节点的分支数目的 2 倍. () (2024)



红黑树性质与特点 (5 道题)

1. 找一个应用场景, 举例说明红黑树有但 AVL 树不具有的优势. (2019)
2. 红黑树中所有节点的黑深度和黑高度之和相等. () (2020)
3. 红黑树某一条路径上有 5 个红节点, 则该红黑树最少有 个节点. (2022)
4. 关键码数量同为 2023 的 AVL 树和红黑树各做一次搜索, 在最坏情况下后者的搜索长度比前者更长. () (2023)
5. 某红黑树有 10 个黑色的节点, 则最多有 个红色的节点. (2025)

红黑树插入与删除 (1 道题)

1. 若 rb-tree 删除导致双黑修正进行了 $\Omega(\log n)$ 次重染色, 则至少旋转一次. () (2022)



散列 (14 道题)

计算平均查找长度 (3 道题)

1. 散列 (2016) (原题数据没回忆起来, 用类似的题代替)
散列表长为 13, 采用双散列函数解决冲突: $H(\text{key}) = \text{key} \% 13$, $H'(\text{key}) = (7 \times \text{key} \% 10) + 1$. 关键码的插入



顺序为:12,23,45,57,20,3,78,31,15,36. (要注意%运算和 $\times$, $\div$ 运算的优先级是一样的.)

- 1)构造散列表.
- 2)求等概率下搜索成功的平均查找长度.
2. 采用独立链 (Scparete chaining) 法的某散列表长度为 119,在存在 2023 个词条时,一次成功查找的过程中所需试探次数的期望值是____. (2023)
3. 散列表长是 7,散列函数用除余法,用线性试探的方式解决冲突,依次插入 55,34,21,13 和 8,插入过程中未进行重散列,如果等概率地查找这 5 个数,则对于一次成功查找期望试探____个桶单元.(2025)

开散列和闭散列的特点 (2 道题)

1. 与开放散列相比,封闭散列有什么优点? 试举例说明两点. (2019)
2. 开放式散列与封闭式散列相比可以更有效地利用缓存. () (2020)



双平方探测 (3 道题)

1. 对长度为 $m=4k+3$ 形式的素数的散列表,采用双平方探测一定能访问其全部元素. () (2018)
2. 采用双平方试探策略,散列表长应取为模 4 于 1 的素数. () (2024)
3. 一个初始为空容量为 $M=13$ 的哈希表 H,采用除留余数法寻址,采用双向平方试探法解决冲突,现插入如下数据,按格式填写下表 (空桶不做标记) (2022)

H[]	0	1	2	3	4	5	6	7	8	9	10	11	12
插入 44													
插入 35													
插入 18													
插入 58													
插入 71													
插入 32													
插入 84													

如果此时插入 2021 会有什么效果?为什么?

懒惰删除 (2 道题)

1. 某散列表长为 2017,采用单平方探测法,已存入 1000 个元素,此时最多有 () 个懒惰删除的桶单元. (2018)
 (A)8 (B)9 (C)1016 (D)1017
2. 某规模为 M 的散列表使用线性试探的方式解决冲突,现共有 N 个词条和 L 个懒惰删除标记位,则执行一次失败的删除的期望时间复杂度是 (). (2024)
 (A) $O\left(\frac{M}{M-N-L}\right)$ (B) $O\left(\frac{M}{M-N+L}\right)$ (C) $O\left(\frac{M}{M-N}\right)$ (D) $O\left(\frac{M-N}{N-L}\right)$

其他散列问题 (1 道题)

1. 若 M 取不超过散列表长度的素数,则不能保证存储的关键码分布均匀. () (2017)



跳转表 (3 道题)

1. 在 n 个词条的跳转表中, 单个词条的期望塔高是 $\Theta(\log n)$. () (2020)
2. n 个词条的 skiplist 期望高度为 () (2024)
(A) $O(n)$ (B) $O(\log n)$ (C) $O(1)$ (D) $O(n \log n)$
3. 有 n 个词条的跳转表 (skiplist) 所占用的空间了, 在极端情况下会大于 $\Omega(n^2)$. () (2023)

图 (共 12 道题)

BFS 的 $\text{dist}(s)$ 相关结论 (1 道题)

1. 对于无向图, 从任意顶点出发, 进行 BFS, BFS 辅助队列中的节点与起始节点距离相差均不大于 1. () (2020)



DFS 标记边类型 (2 道题)

1. DFS 过程中何时标记前向边? 何时标记后向边? (2019)
2. 有向图经过 DFS 后有 k 条边被标记为后向边, 则它不一定恰有 k 个环路. () (2020)

BCC (1 道题)

1. 以无向图某一顶点 s 为起点做 DFS, 若生成的 DFS 树中 s 的度数为 1, 则 s 必不为关节. () (2022)

Dijkstra 算法 (2 道题)

1. 边的权值都为正整数的图可以用 Dijkstra 算法构造出最短路径. () (2016)
2. 只要存在负权边, 则 Dijkstra 算法即使能够终止, 所输出的也不一定是最短路径. () (2023)

最小生成树, Prim 算法, Kruskal 算法 (2 道题)

1. 用 V 表示图的顶点集, U 表示已经确定路径的顶点集, 有如下算法: 初始时 U 为空, F 为已经确定的路径集合, 初始也为空. 先任意取一点 u 放入 U , 然后在 $V-U$ 中遍历 u 的邻接点, 选权值最小的边 e 和点 v 放入 F 和 U 中. 该算法是否能够构成最小生成树? 如果能则给出证明, 如果不能举出例子. (2016)
2. 相对于 Kruskal 算法, Prim 算法更适宜于稀疏图. () (2023)



MST 数量问题 (1 道题)

1. ABCD 四个顶点构成的完全图, 只有一条边的权值是 2, 其他边权值是 1, 则该图对应的 MST 共有 ____ 棵. (2025)



拓扑排序 (3 道题)

1. 有向无环图的拓扑排序是 DFS 的 () (2020)
(A) 被发现的顺序 (B) 被发现的逆序 (C) 回溯的顺序 (D) 回溯的逆序

2. 深度优先搜索+拓扑排序 (2021)

对任一有向无环图做 DFS, 所有节点按照回溯的次序逆序排列, 是否总是一个拓扑排序? 如果是, 试予证明; 否则, 试给出一个反例.

3. 存在拓扑排序的有向图未必无环. () (2024)



优先级队列 (15 道题)

完全二叉堆的插入和删除 (3 道题)

1. 随机输入的情况下, 完全二叉堆的插入的平均时间复杂度是 $O(1)$, 虽然最坏是 $O(n)$. () (2018)2. 在理想随机情况下, 完全二叉堆插入操作的平均时间复杂度只需 $O(1)$, 尽管最坏情况下为 $O(\log n)$. () (2019)3. 完全二叉堆删除元素在最坏情况下时间复杂度为 $O(\log n)$, 但是平均情况下为 $O(1)$. () (2020)

Floyd 建堆 (3 道题)

1. 建立一个完全二叉堆时间复杂度为 $O(n \log n)$. () (2020)

2. 计算 (10 分) (2023)

试采用 Floyd 算法将以下向量转换为大顶堆, 并按如下格式, 依次给出向量在各次下滤之后的内容, 相对于上一行没有变化的元素直接留白, 不要重复填写.

K	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
向量 A[k]	3	8	0	6	1	13	11	2	9	5	4	14	7	10	12
第 1 次下滤后															
第 2 次下滤后															
第 3 次下滤后															

3. $\{12, 29, 20, 24, 70, 80, 150\}$ 用 Floyd 建堆, 建大根堆, 结果为 $\{150, _, _, _, _, _, _ \}$ (2025)

胜者树、败者树 (3 道题)



1. 与锦标赛树相比, 败者树的优势是什么? (2019)

2. 败者树删除的时间复杂度在常数上优于胜者树. () (2020)

3. 使用胜者树可以在 $O(n \log n)$ 时间内完成堆的 n 个整数排序, 败者树不行. () (2024)

左式堆 (5 道题)

1. 对于一个右侧路径长度为 k 的左式堆, 其节点数量 () 为 () (2018)

(A) 至少; 2^k (B) 至少; $2^{k+1} - 1$ (C) 至多; 2^k (D) 至多; $2^{k+1} - 1$

2. 采用 crane 算法将左式堆 A 和 B 合并为左式堆 H, 则 H 右侧链上的节点未必都来自 A 或者 B 的右侧链. () (2020)

3. 左式堆兄弟子堆满足 $L.height \geq R.height$. () (2022)

4. 在包含 2023 个节点的左式堆中, 右子堆最多可能有 _____ 个节点 (2023)

5. 左式堆能兼顾 $\text{merge}()$, $\text{delmax}()$ 和 $\text{insert}()$ 三个接口的时间复杂度, 使三者都为 $O(\log n)$. () (2024)

多叉堆 (1 道题)

1. Dijkstra 用于稠密图时, 为何使用多叉堆替换掉完全二叉堆? 多叉堆的分叉数怎么确定? (2019)



串 (共 16 道题)

KMP 复杂度 (4 道题)

1. 利用没改进的 $\text{next}[]$ 表, KMP 算法时间复杂度也是 $O(n+m)$. () (2017, 2018, 2019)
2. 使用未改进的 $\text{next}[]$ 表的 KMP 算法不能保证 $O(n+m)$ 的时间. () (2025)

$\text{next}[]$ 表, 改进版 $\text{next}[]$ 的构造 (3 道题)

1. 模式串 HHFBHHFHHFBSHF 的改进后 $\text{next}[]$ 表中 $\text{next}[13]$ 和 $\text{next}[1]$ 的值 () (2020)
(A) $\text{next}[13]=1, \text{next}[1]=-1$ (B) $\text{next}[13]=1, \text{next}[1]=0$
(C) $\text{next}[13]=0, \text{next}[1]=-1$ (D) $\text{next}[13]=-1, \text{next}[1]=-1$
2. 试按如下格式, 分别给出下面模式串对应的 $\text{next}[]$ 表和改进后的 $\text{next}[]$ 表: (2021)

j	0	1	2	3	4	5	6	7	8	9	10	11	12
p[j]	R	V	M	M	O	R	E	O	T	R	V	M	P
$\text{next}[j]$	-1												
改进 $\text{next}[j]$	-1												

3. KMP 算法中, 模式串“BABABBBBABABBABBA”的改进版 $\text{next}[]$ 表最大项与最小项的差为 _____. (2024)



KMP 和 BF 算法的比较 (3 道题)

1. 在字符集中的各字符出现概率相同时, KMP 算法渐进时间复杂度接近蛮力算法. () (2017)
2. KMP 算法相对于蛮力算法的优势在什么条件下足够明显? 为什么? (2019)
3. 模式串和文本串均由 26 个大写字母随机组成, 那么在最好情况下, 蛮力算法的时间复杂度 () KMP 算法, 在平均情况下, 蛮力算法的时间复杂度 () KMP 算法. (2020)
(A) 相等, 大于 (B) 大于, 大于 (C) 等于, 大于 (D) 等于, 等于

bm 算法 (5 道题)

1. 对于长度为 m 的随机 0,1 串进行串匹配时, 好后缀数组中 $\text{gs}[0]=1$ 的概率为 () (2018)
(A) $\frac{1}{m}$ (B) $\frac{1}{2^{m-1}}$ (C) $\frac{1}{2^m}$ (D) $\frac{1}{2^{m+1}}$
2. 如果 bm 算法仅使用好后缀策略而不使用坏字符策略则不能保证在最好情况下有 $O(n/m)$ 的时间复杂度 (其中 n 为文本串长度, m 为模式串长度) () (2022)
3. 设文本串 T 和模式串 P 的长度分别为 2023 和 17, 且 T 并不包含 P , 借助 Boyer-Moore 算法的坏字符表 bc (好后缀表 gs 不用) 来做模式匹配, 最好情况下只需做 ____ 次字符比对. (2023)
4. 在 BM 算法中坏字符表的长度与模式串长度无关. () (2024)
5. 仅采用 $\text{bc}[]$ 策略的 BM 算法, T 串长 2025, 只有 0 和 1, P 串为“x20241222”, 最好的情况下仅需要比对

____次. (2025)



Karp-Rabins 算法 (1 道题)

1. Karp-Rabin 算法在匹配指纹失败后无法在 $O(1)$ 时间内计算出下一个指纹. () (2022)

排序 (共 24 道题)

起泡排序 (1 道题)

1. 在执行起泡排序时,有可能出现某个元素在排序过程中一直远离它的最终位置的情况 () (2016)



插入排序,折半插入排序 (4 道题)

1. 折半插入排序算法在寻找插入位置时采用二分查找,因此折半插入排序的时间复杂度为 $O(n \log n)$ () (2016)
2. 对序列 $(64, 63, \dots, 2, 1)$ 进行直接插入排序,总的比较次数最接近于 () (2017)
(A) 2800 (B) 2600 (C) 2400 (D) 2200 (E) 2000
3. 与选择排序相比,插入排序有什么优点?试举例说明两点. (2019)
4. 在输入数据是逐个给出的应用场景中, () 排序算法比较好. (2024)
(A) 插入排序 (B) 选择排序 (C) 归并排序 (D) 快速排序



快速排序 (3 道题)

1. 快速排序平均时间复杂度为 $O(n \log n)$,最好时间复杂度也是 $O(n \log n)$. () (2020)
2. 快速排序和插入排序 (7 分) (2021)
考察对 QuickSort 算法的如下调整:任何子序列在规模缩减至 m 以下之后,便不再继续递归划分;待所有递归都已终止后,再对整个序列调用一次 InsertionSort,如果待排序序列总长为 n ,那么 InsertionSort 需要花费多少时间? 为什么?
3. 元素 $\{1, 2, \dots, 2025\}$ 随机构成的序列,进行快速排序,则 212 与 2000 进行比较的概率为 _____. (2025)

希尔排序 (2 道题)

1. 若步长序列采用 $\{1, 3, 7, 15, \dots, 2^k - 1, \dots\}$,则希尔排序的时间复杂度是 $O(n^{1.5})$. () (2022)
2. 记非负整数集为 N ,采用增量序列 () 的希尔排序需做 $\Omega(\log^2 n)$ 趟排序. (2024)
(A) $\{2^p | p \in N\}$
(B) $\{2^{p+1} - 1 | p \in N\}$
(C) $\{2^p \cdot 3^q | p, q \in N\}$
(D) $\{9 \times 4^p - 9 \times 2^p + 1 | p \in N\} \cup \{4^p - 3 \times 2^p + 1 | p \in N_+\}$



排序算法的稳定性,就地性 (3 道题)

1. 插入排序是就地 (in-place) 算法. () (2023)
2. 稳定的排序算法,能够保持等值元素在原输入中的相对次序. () (2024)

3. 基于向量的起泡排序不是就地算法. () (2024)



k-选取问题 (2 道题)

1. 基于比较式的算法可以在 $O(n)$ 内确定无序数据中的前 10%. () (2020)
2. CBA 式算法可以在 $O(n)$ 内从任意 n 个整数中找出中位数. () (2025)

CBA 式排序算法复杂度问题 (1 道题)

1. 采用 CBA 式的排序算法对所有大小为 n 的数组进行排序,其时间复杂度是 $\Omega(n \log n)$. () (2018)

循环节 (2 道题)

1. 在插入排序过程中,有序前缀每增加一个元素,序列中的循环节个数即使不增加,亦不致减少 () (2019)
2. 一个序列中的两个元素分别属于不同的循环节,则他们交换后,循环节总数未必减少. () (2025)

逆序对 (2 道题)

1. 交换序列中的任一对逆序元素,该序列的逆序对的个数必然减少. () (2019,2025)

基数排序 (3 道题)



1. 基数排序的底层排序算法一定是稳定的. () (2018)
2. 基数排序如果采用了不稳定的底层排序算法,则输出序列未必正确. () (2019)
3. 采用不稳定的底层排序算法的基数排序 () (2020)
(A)未必正确且不稳定 (B)正确并且不稳定 (C)未必正确并且稳定 (D)正确并且稳定

其他问题 (1 道题)

1. 随机整数组成的一个矩阵,先进行行排序,后进行列排序,之后各行必然仍然有序,这里有序指单调非降. () (2025)

算法题 (13 道真题+6 道补充)

向量 (2 道真题+3 道补充)

1. 判断子数组 (2016)

求一个数组 $A[]$ 中连续相同数字的和等于 s 的最长子数组长度,例如 $A=\{1,1,2,1,1,1,2,1\}$, $s=3$,则所求子数组长度为 3,要求算法时间复杂度不超过 $O(n)$,空间复杂度不超过 $O(1)$

- 1)描述算法思想
- 2)伪代码实现
- 3)计算程序的算法复杂度.



2. 连续子序列的最大和 (2018)

求一个整数类型的向量 $A[0,n)$ 中最大的连续子序列的和.

1) 说明你的算法.

2) 使用伪代码描述你的算法.

3) 分析你算法的空间和时间复杂度. (注: 你的得分取决于你的算法性能, 性能在 $O(n^2)$ 以内的才能得分, 时间复杂度 $O(n)$ 的才有可能得满分, 蛮力算法不得分, 因为其时间复杂度为 $O(n^3)$)

例如: 输入的数组为 1,-2,3,10,-4,7,2,-5, 和最大的子数组为 3,10,-4,7,2, 因此输出为该子数组的和 18.

3. 有序向量中是否有和为 s 的两个数 (补充同类型题目 1)

[167. 两数之和 II - 输入有序数组 - 力扣 \(LeetCode\)](#)

给你一个下标从 1 开始的整数数组 $numbers$, 该数组已按非递减顺序排列, 请你从数组中找出满足相加之和等于目标数 $target$ 的两个数. 如果设这两个数分别是 $numbers[index1]$ 和 $numbers[index2]$, 则 $1 \leq index1 < index2 \leq numbers.length$.

以长度为 2 的整数数组 $[index1, index2]$ 的形式返回这两个整数的下标 $index1$ 和 $index2$.

你可以假设每个输入只对应唯一的答案, 而且你不可以重复使用相同的元素.

你所设计的解决方案必须只使用常量级的额外空间.

4. 统计逆序对个数 (补充同类型题目 2)

[LCR 170. 交易逆序对的总数 - 力扣 \(LeetCode\)](#)

在股票交易中, 如果前一天的股价高于后一天的股价, 则可以认为存在一个「交易逆序对」. 请设计一个程序, 输入一段时间内的股票交易记录 $record$, 返回其中存在的「交易逆序对」总数.

示例:

输入: $record = [9, 7, 5, 4, 6]$

输出: 8

解释: 交易中的逆序对为 (9, 7), (9, 5), (9, 4), (9, 6), (7, 5), (7, 4), (7, 6), (5, 4).

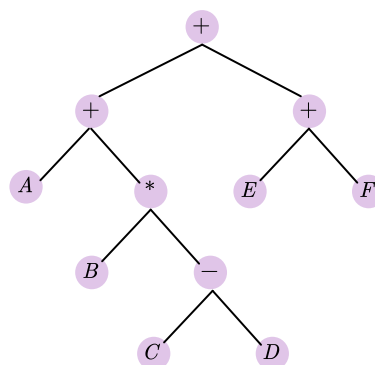
5. 找众数 (补充同类型题目 3)



栈 (1 道真题+1 道补充)

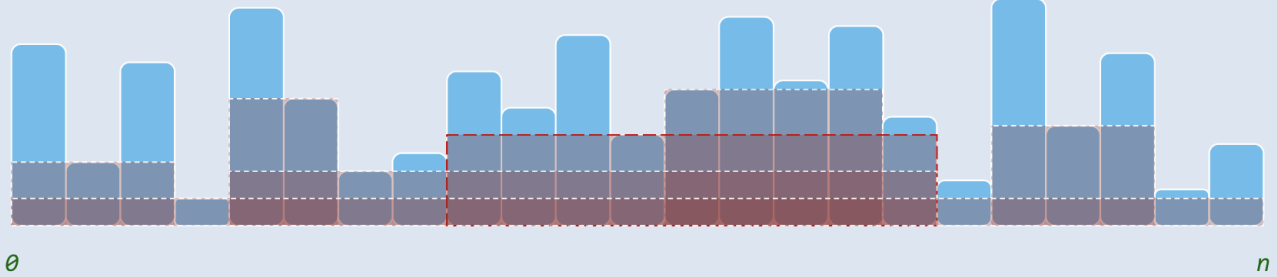
1. 构造表达式树 (2016)

设计一个算法, 根据表达式树的后序遍历序列构造出对应的表达式树, 例如 $ABCD-*+EF++$ 构造出如图所示的二叉树.



2. 最大直方图 (补充题 1)

Maximum Rectangle



❖ Let $H[0,n)$ be a histogram of non-negative integers

❖ How to find the largest orthogonal rectangle in $H[]$?

❖ To eliminate possible ambiguity

we can, for example, choose the **LEFTMOST** one

[LCR 039. 柱状图中最大的矩形 - 力扣 \(LeetCode\)](#) (单调栈问题之一)

二分查找 (2 道真题+2 道补充)



1. 单峰向量 (2018)

向量 $A[0,n)$ 为单峰向量须满足其前缀严格递增 $A[0,k)$, 其后缀 $A[k,n)$ 严格递减.

- 1) 设计算法在 $O(\log n)$ 的时间内找到单峰向量 $A[0,n)$ 的最大值所在的秩 k .
- 2) 证明你算法的正确性.
- 3) 证明即使在最坏的情况下, 你的算法复杂度也不会超过 $O(\log n)$.

leetcode 有官方题解: [852. 山脉数组的峰顶索引 - 力扣 \(LeetCode\)](#)

2. 求二叉树的后序遍历序列中的第 k 个节点 (7+4+3=14 分) (2019)

```
typedef struct BinNode{
    int size;           //当前节点和孩子总数
    struct BinNode *lc,*rc; //左右孩子指针
}BinNode;

BinNode* rank(BinNode* t,unsigned int k){
    //0<=k<= t.size 在此处补充不超过 12 行的代码
    //rank(t,k)实现以 t 为根的二叉树中按照后序遍历顺序找出第 k 个节点
    //若目标节点为 x,则算法时间空间复杂度均不可以超过  $O(\text{depth}(x))$ ,即节点深度
    //请不要直接模拟后序遍历,复杂度会超过要求,将不得分
}
```

(1)给出具体实现算法

(2)说明原理 (200 字以内,可附一张插图)

(3)证明时间空间复杂度 (不超过 120 字)

3. 找两个有序向量归并之后的中位数 (补充同类型题目 1)

4. 搜索二维矩阵 (补充同类型题目 2)

编写一个高效的算法来搜索 $m \times n$ 矩阵 `matrix` 中的一个目标值 `target`. 该矩阵具有以下特性: 每行的元素从左到右升序排列, 每列的元素从上到下升序排列.

示例:

1	4	7	11	15
2	5	8	12	19
3	6	9	16	22
10	13	14	17	24
18	21	23	26	30



输入:matrix=[[1,4,7,11,15],[2,5,8,12,19],[3,6,9,16,22],[10,13,14,17,24],[18,21,23,26,30]], target = 5

输出:true

[240. 搜索二维矩阵 II - 力扣 \(LeetCode\)](#)

二叉树或者 BST 问题 (7 道真题)

1. 二叉树的遍历 (2017)

若二叉树的数据结构如下

```
struct binarytree{
    struct binarytree *parent;
    struct binarytree *lc;
    struct binarytree *rc;
    struct binarytree *first();
};
struct realbinarytree{
    struct binarytree p;
    struct binarytree *next();
};
```



- (1)若 first()函数是返回二叉树后序遍历节点的第一个节点,请写出 first()函数代码.
- (2)若 next()函数是返回该节点的后序遍历的后继节点,请写出 next()函数代码.
- (3)在调用 first()函数和 next()函数对二叉树进行后序遍历时,证明其算法时间复杂度为 $O(n)$.

2. 算法题 (A,B,C,D 和 F 都是 3 分每题,E 为 5 分.) (感觉回忆的题目有点怪)

下图给出二叉树的结构体声明:

```
struct BinNode{ //二叉树节点
    BinNode *lc;
    BinNode *rc;
    BinNode *parent;
    int size;
};
```

- A.完全二叉树左子树的规模为_____,请给出递推公式;
- B.给出 A 的伪代码实现 lSize();
- C.search(k)找出中序遍历序列第 k 个节点 (中序遍历起始下标为 0,不要尝试模拟中序遍历,模拟中序遍历将不得分) ;
- D.splyto(a, x) ,a 为 x 的祖先,通过 zig-zag 操作将 x 调整变为 a 的孩子,若 a 为 NULL,x 调为根节点;
- E.将一棵 splay 树调整成完全二叉树 (可用上述函数),要求时间复杂度为 $O(n\log n)$,迭代深度不超过 $O(\log n)$
- F.证明你在 E 中给出的算法满足复杂度和迭代深度的要求.

3. 算法设计与分析 (8+5+3=16 分) (2021)

设二叉搜索树(Binary Search Tree)的节点定义如下:



```
struct BstNode{
    int key;           //关键码
    BstNode *lc;       //左孩子
    BstNode *rc;       //右孩子
    BstNode *parent;   //父节点
    int size;          //当前节点及其后代的总数
}
```

(1)试用伪代码完成以下算法,并提供足够的注释说明:

```
int count(BstNode *T, int lo, int hi){
    //在以 T 为根的二叉搜索树中,统计出在[lo,hi)区间内关键码的总数;
    //无论统计出的总数有多大,时间复杂度均不得超过 O(h),其中 h 为全树高度;
    //除了作为输入的这棵树,只能使用 O(1)的辅助空间;
}
```

(2)试说明算法的原理及正确性.

(3)试证明算法的时间,空间复杂度均符合要求.

4. 算法题 (2022)

已知任何一棵多叉树 M 可以通过长子-兄弟法表示为二叉树 T ,二叉树的层次遍历法也可用于多叉树,现给出二叉树节点定义如下:

```
struct BinNode{
    int data
    BinNode *lc
    BinNode *rc
};
```

请设计如下要求的遍历函数,并配合必要的注释:

```
void traverse(BinNode *x){
    //x 是二叉树 T 的根节点,T 对应了一棵多叉树 M 的长子-兄弟表示法
    //要求输出全部节点的数值,遍历次序等价于 M 的层次遍历次序
    //不可以使用递归,也不可以改变 T 的结构
    //不得使用队列和栈以外的辅助数据结构
    //时间和空间复杂度皆为 O(n),n 为节点数量
}
```



(1)实现代码.

(2)说明该算法的原理.

(3)证明该算法的正确性.

(4)证明该算法的时间、空间复杂度为 $O(n)$.



5. 算法题 (7+10+3=20 分) (2023)

以 r 为根的二叉树中,任一节点 v 与 r 之间通路上的所有节点 (包括 r 及 v 本身) 都成为 v 的祖先. 如果节点 a 同时是节点 v 的和 u 的祖先,则称作 v 和 u 的公共祖先 (当然 r 是所有节点的公共祖先) 其中的深度最大者,称作最低公共祖先 (Lowest Common Ancestor) .

假定存有 n 个节点的二叉树中,各节点的关键码都是取自 $[0,n)$ 的正整数,且彼此互异,尽管不能直接访问该树,但其先序、后序遍历序列已分别记录在数组 $P[0,n)$ 和 $Q[0,n)$ 中,试设计一个算法对 $[0,n)$ 内的任意一对整数关键码 x 和 y ,找出它们所对应节点的最低公共祖先 $LCA(x,y)$,要求算法的运行时间和辅助空间均不超过 $O(n)$.

比如 $P=\{3,0,8,5,2,6,7,10,4,1,9\}$, $Q=\{5,8,2,0,10,4,7,9,1,6,3\}$, 则 $LCA(2,5)=0, LCA(8,4)=3, LCA(6,9)=6$

(1) 试通过文字与图示, 说明算法的原理.

(2) 试用伪代码描述算法的过程, 并提供足够的注释.

(3) 试证明算法的时间, 空间复杂度均符合要求



6. 算法题 (2+2+8+4+2+2=20 分) (2024)

所谓的有根有序树 T , 都有唯一的根节点 $r=\text{root}(T)$, 且子树可编号为 $\{T_1, T_2, \dots, T_d\}$. 子树总数 d 亦称作 r 的度数. 于是, 可递归定义 T 的先序遍历序列如下: $\text{pre}(T)=\{r\}+\text{pre}(T_1)+\text{pre}(T_2)+\dots+\text{pre}(T_d)$ 其中“+”为序列串接运算. 类似可定义后序遍历序列, 如下: $\text{post}(T)=\text{post}(T_1)+\text{post}(T_2)+\dots+\text{post}(T_d)+\{r\}$. 问题:

(1) 已知某棵有根有序树 T 的节点用整数编号, 其先序遍历序列为 $\text{pre}(T)=\{9,2,3,4,1,5,8,7,6,0\}$ 对应的节点度数序列, 按上述次序给出了各节点的度数: $\text{deg}\{T\}=\{3,2,0,0,1,3,0,0,0,0\}$ 试画出该树的结构 (子树必须自左而右“有序”).

(2) 试给出该树的后序遍历序列 $\text{post}(T)$.

(3) 试用伪代码结合足够的注释, 描述如下算法: $\text{pre2post}(\text{stack}\langle\text{int}\rangle\&\text{pre}, \text{stack}\langle\text{int}\rangle\&\text{deg}, \text{stack}\langle\text{int}\rangle\&\text{post})$; 任给一棵有根有序树的先序遍历序列, 以及对应的节点度数序列, 计算出对应的后序遍历序列. 初始时, 先序遍历序列, 节点度数序列分别自顶而下地存放在栈 pre, deg 中, post 栈为空; 算法终止时, pre, deg 栈都应为空, 后序遍历序列则应自底而上存放于 post 栈中. 算法运行时间, 应线性正比于树的规模. 算法过程中任何时刻的空间占用量 (含调用栈) 相对于初始时的空间占用量不超过 $O(1)$.

(4) 试证明算法的正确性.

(5) 试证明算法的时间复杂度符合要求.

(6) 试证明算法的空间复杂度符合要求.

7. 算法题设计与分析 (7+1+5+1=14 分) (2025)

```
Struct BinNode{
    int data; // 数据
    BinNode *lc, *rc; // 左右孩子
    BinNode *parent; // 父亲 (对根结点而言是 NULL)
}
```



均由 n 个节点组成的两棵二叉树 V 和 U , 若其节点可以分别用 1 到 n 之间的整数编号, 且每一对编号相等的节点 $v \in V$ 和 $u \in U$, 都满足 $v \rightarrow \text{parent}$ 和 $u \rightarrow \text{parent}$ 均为 NULL 或者编号相等, 则称 v 与 u 弱同构, 进一步地, 若还满足 $v \rightarrow \text{lc}$ 和 $u \rightarrow \text{lc}$ 均为 NULL 或者编号相等且 $v \rightarrow \text{rc}$ 和 $u \rightarrow \text{rc}$ 均为 NULL 或者编号相等, 则称 V 与 U 强同构

(1) 完成以下算法

```
// 以 x 为根的二叉树中, 共有 data 域互异的 n 个节点
void heapify(BinNode *x){
    // 在 O(nlogn) 时间内, 将其转化为与之强同构的大根堆
}
```



(2) 试说明你的 $\text{heapify}()$ 算法时间复杂度满足要求

(3) 接下来递归地定义二叉树节点的 npl 值如下:

对于空节点 NULL, $\text{npl}(\text{NULL})=0$

对于非空节点 v , $\text{npl}(v)=1+\min\{\text{npl}(v \rightarrow \text{lc}), \text{npl}(v \rightarrow \text{rc})\}$

若堆中的每个非空节点 v , 都满足 $npl(v)=1+npl(v \rightarrow rc)$, 则称之为左式堆

完成如下算法:

//以 x 为根的大顶堆中, 共有 $data$ 域互异的 n 个节点

```
void leftify(BinNode *x){
```

```
    //在  $O(n)$  时间内, 将其转化为与之弱同构的左式堆
```

```
}
```

(4) 试说明你的 `leftify()` 算法时间复杂度满足要求.



图 (1 道真题)

1. 广搜最小环 (2017)

请利用图的广度优先遍历找出图中的最小环的长度, 若不存在环, 则输出 $+\infty$, 要求时间复杂度为 $O(ne)$, 空间复杂度为 $O(n)$, 最小环即环中边数最少的环.

1) 请描述你的算法思想.

2) 请用伪代码写出算法.

3) 说明你的算法的时间复杂度和空间复杂度.



组成原理

绪论 (8 道题)



冯诺依曼机 (3 道题)

1. 计算机硬件执行程序的最小单位是 () (2017)
(A) 公众号:新威考研 (B) 微指令 (C) 指令 (D) xinweikaoyan.com
2. 冯诺依曼体系计算机把指令和数据存放在存储器中. () (2019)
3. 冯诺依曼计算机的最主要特征是____. (2021)

主频,CPI,时钟周期,执行时间的相互关系 (5 道题)

1. 提高 CPU 主频可以加快程序执行速度,CPU 的主频越高,指令执行就越快. () (2018)
2. CPI 越小,执行程序用时就越短. () (2019,2020)
3. 执行相同的程序,以四个处理器所需要的时间最短的是 () (2021)
(A) 处理器 1 频率为 3GHz, CPI=1.5 (B) 处理器 2 频率为 2.5GHz, CPI=1.0
(C) 处理器 3 频率为 4GHz, CPI=2.2 (D) 处理器 4 频率为 2.8GHz, CPI=1.1
4. 计算机甲的 CPI=3, 计算机乙的 CPI=1, 执行相同的程序, 乙一定快于甲. () (2024)

数据的表示与计算 (共 29 道题)

补码多一个负数的问题 (3 道题)

1. C 语言中, int x, y 若有 $x > y$ 成立, 则一定有 $-x < -y$ 也成立. () (2018)
2. C 语言中 int x, 若 $x < 0$ 成立, 则必有 $-x > 0$ 也成立. () (2020)
3. 若 x 和 y 是补码, 则 () (2024)
(A) 若 $-x < 0$ 则 $x > 0$ (B) 若 $-x > 0$ 则 $x < 0$ (C) 若 $-x < -y$ 则 $x > y$ (D) 若 $-x > -y$ 则 $x < y$



具体数据的补码表示 (5 道题)

1. -2017 的 32 位补码表示为____. (16 进制或者 2 进制表示) (2018)
2. +1234 的 32 位补码在内存中存储的信息为 0x____. (用 16 进制表示, 假设采用小端存储) (2019)
3. 采用补码表示的 short 类型数据, 存放在内存中的连续 2 个字节单元, 其值分别为 0xFF (低地址所在字节) 和 0x01 (高地址所在字节), 如果是小端机, 则其十进制值为____, 如果是大端机, 则其十进制值为____. (2021)
4. 补码表示的一个 4 字节的有符号整数的最大值是 0x____. (16 进制表示) (2022)
5. 计算机中的二进制数 1100 0000 1110 0000 0000 0000 0000 0000 (右侧为低位) 按原码表示的十进制无符号原码是____, 按补码表示法表示的十进制整数是____. (2025)

补码加法满足交换律 (1 道题)

1. 用补码表示的 32 位正数加法满足交换律. () (2023)



其他补码问题 (1 道题)

1. 若 x 是 C 语言中的 `int` 型变量, 如果 $x > 0$ 成立, 则 $x \cdot x > 0$ 也成立. () (2019)

负数的原码, 补码, 反码的关系 (1 道题)

1. 负数的原码, 补码, 反码都不相同. () (2024)



IEEE754 浮点数的临界值 (3 道题)

1. IEEE754 规格化单精度浮点数所能表示的最小正数是 () (2017) (失忆的选项由已修复)
(A) 2^{-126} (B) 2^{-127} (C) 2^{-148} (D) 2^{-149}
2. IEEE754 单精度浮点数标准下, 最大规格化数在内存中表示为 $0x$ _____. (16 进制表示) (2022)
3. 在 IEEE754 单精度浮点数标准下, 最小非规格化正数的十六进制表示为 $0x$ _____. (2024)

具体数据的 IEEE754 浮点数表示 (3 道题)

1. -2017 的 IEEE754 单精度浮点表示为 _____. (16 进制或 2 进制表示) (2018)
2. -27.625 在 IEEE754 单精度浮点数下表示为 $0x$ _____. (用 16 进制表示) (2019)
3. 计算机中的二进制数 1100 0000 1110 0000 0000 0000 0000 0000 (右侧为低位), 按 IEEE754 表示的单精度浮点数是 _____ (结果可用 2 的 k 次方的和表示). (2025)



其他浮点数问题 (4 道题)

1. 浮点数 `float y`, 若 $y * y$ 仍然在浮点数表示范围内, 则 y 大于 0. () (2020) (严重怀疑问的是 $y^2 > 0$)
2. 关于 IEEE754 单精度浮点数描述正确的是 () (2021)
(A) 正数的个数比负数的个数多 (B) 负数的个数比正数的个数多
(C) 正数的个数比负数的个数一样多 (D) 依据实现有不同的情况
3. IEEE754 双精浮点数加法满足结合律. () (2023)
4. 在 IEEE754 标准浮点表示中, 阶码采用二进制补码表示. () (2025)

非数字编码 (3 道题)

1. 在字符的编码中包含了字符如何显示的信息. () (2021)
2. 以下关于字符描述的说法中错误的是 () (2023)
(A) ASCII 码使用一个字节进行编码 (B) UTF-8 码使用两个字节进行编码
(C) 字符编码不包括字符如何显示的信息 (D) 使用矢量字体的字符放大后不会失真
3. 在 UTF-8 编码中任意读取一个字节, 一定能判断它是否为英文字符. () (2024)



逻辑电路和 ALU 问题 (2 道题)

1. 处理机____逻辑电路进行算术运算,____逻辑电路用于数据暂存,____逻辑电路用于分支选择. (2018)
2. 关于 ALU 超前进位的描述正确的是 () (2022) (回忆不起的选项已被新威考研强行修复)
(A)简化电路结构 (B)可以更快地获得进位 (C)可以有更高的精度 (D)可加快逻辑运算速度

海明码 (3 道题)

1. 若海明码 P1P2D1P3D2D3P4 为 0101100,则该海明码有____位错误 (0 位,一位,二位),正确的 D1D2D3 为____. (2017)
2. 三位二进制数据 101 (D1D2D3) 的对应的带全局校验的海明码为____. (2023)
3. 海明码编码的数据位数 k,校验位数为 r,如果支持单检错单纠错,k 和 r 的关系是____.如果有数据 1001,则单检错单纠错编码,其编码为____. (2025)

指令系统 (共 5 道题)

指令基本概念相关问题 (2 道题)

1. 指令是由____和____组成. (2017)
2. 下列表述中正确的是 () (2020)
(A)指令必须给出指令操作码 (B)指令必须给出指令操作数
(C)指令的长度必须一样长 (D)指令的操作数的个数都一样多



RISC 和 CISC (1 道题)

1. 关于 X86 和 MIPS 的描述正确的是 () (2022)
(A)x86 指令集属于 RISC 指令集 (B)x86 指令格式规整且等长
(C)MIPS 指令集属于 RISC 指令集 (D)MIPS 指令支持间接寻址方式



计算指令长度 (1 道题)

1. 指令系统操作数地址占 6 位,有一地址 14 条,二地址 253 条,三地址 20 条,指令的长度至少为 () (2024)
(A)23 (B)24 (C)26 (D)32



寻址方式 (1 道题)

1. 在 RISC-V 典型五级流水线设计,执行一段指令序列
lw x11,16(x10); //从内存中读取一个字符到寄存器 x11 中.
add x13,x11,x12; //将寄存器 x11 和 x12 值相加,结果置于寄存器 x13 中
addi x14,x13,1; //将 x13 寄存器的值与 1 相加,结果置于 x14 中

上述指令采用的寻址方式有 3 个____,____,____. (2025)

控制器设计 (共 38 道题)



单周期,多周期,流水线指令延迟的计算 (2 道题)

1. 某 MIPS 计算机,其内存延迟为 10ns,一个周期一次读或者写,ALU 的延迟为 6ns,寄存器堆的输出延迟为 3ns,输入延迟为 1ns,一个时钟周期内可以完成两个寄存器的读操作和一个寄存器的写操作.流水线寄存器及多周期锁存器输出延迟为 2ns,PC 输出延迟为 2ns.每条指令按取指、译码、执行、访存、写回五个阶段执行.实现以下 7 条指令:



```

addu    rd,    rs,    rt
subu    rd,    rs,    rt
ori      rt,    rs,    imm
lw      rt,    rs,    imm
sw      rt,    rs,    imm
J      jarget
  
```

从 PC 的准备输出开始计算 (时钟信号上升沿)

- (1) 按单周期设计,指令内存与数据内存分开,计算指令延迟.
- (2) 按多周期设计,指令内存和数据内存存在同一内存模块,最长和最短的指令延迟分别是哪条指令,分别计算对应的指令延迟.
- (3) 按五级流水线设计,指令内存和数据内存不在同一个内存模块,处理器频率最高能到多少? (2019)

2. 假设寄存器输入延迟为 10ps,寄存器输出延迟为 10ps,各阶段的延迟如下 (其中 ID 段算入了通用寄存器的读取时间,其余未算入) (总感觉这题有点奇怪,回忆的有点模糊,其余段也不读通用寄存器啊,这段括号里的描述相当于说了句废话.) (2020)

指令阶段	IF	ID	EXE	MEM	WB
每段时间	250ps	180ps	150ps	300ps	200ps

实现以下 3 条指令:

```

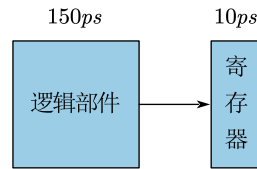
addu    rd,rs,rt
lw      rs,rt,imm
j      target
  
```



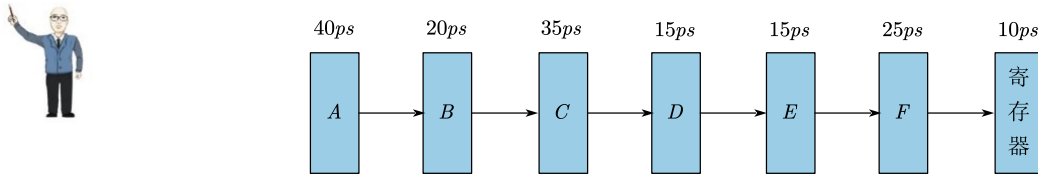
- (1) 按照单周期、多周期、流水线设计,时钟周期最短为?请给出你的计算过程.
- (2) 按照单周期、多周期、流水线设计,最短的指令延迟为?请给出你的计算过程.

流水线段数,段长与吞吐量问题 (3 道题)

1. 任何指令的执行部件可以抽象为一个逻辑部件和寄存器的组合结构,假设逻辑部件的延迟为 150ps,寄存器的延迟为 10ps,示意图如下:



逻辑部件又可以抽象为下图的 A~F 一共 6 个部件的组成,各部件的延迟如下图所示:



- (1) 要形成一个 4 级流水线 CPU,应该将三个寄存器安插在哪些位置会使得吞吐率最大?问该四级流水线的延迟和最大吞吐率.
- (2) 为达到最大的吞吐率应该设计成几级流水线?寄存器应该安插在哪些位置?问该流水线的延迟和最大吞吐率. (2016)

2. 提高流水线的段数可提高 CPU 的频率,并提高单位时间内执行指令的速度. () (2020)

3. 五级流水线处理器实现中,指令流经各个功能段的时间分别为 60ns,50ns,40ns,60ns,30ns 忽略流水线寄存器的延迟,则该处理器的时钟周期至少是 () (2023)

- (A)60ns (B)50ns (C)40ns (D)30ns

流水线基本概念问题 (2 道题)

1. CPU 的流水线寄存器对系统软件程序员是透明的 () (2022)
2. 以下关于五级流水线的处理器说法错误的是 () (2018)
 - (A)多处理器下不会发生结构冲突
 - (B)每条指令的每个时钟周期执行一个功能
 - (C)可以采用微程序或者硬连线设计
 - (D)不同的指令执行时间相同

三种流水线冲突的概念 (2 道题)

1. 指出指令流水线可能发生的冲突分类,以及原因. (2017,简答题)
2. 指令流水线冲突包括结构冲突,数据冲突和____冲突 (2023)

数据冲突和解决数据冲突的方法 (8 道题)

1. MIPS 处理流水线数据冲突的技术有____,____,____. (任写三个即可) (2019)
2. 以下哪种方法不可以解决数据冲突 () (2018,2020,2021)
 - (A)暂停流水线
 - (B)分支预测
 - (C)调整指令顺序
 - (D)数据旁路
3. MIPS 五级流水线中,可能有哪种数据冲突 () (2019)
 - (A)RAR
 - (B)RAW
 - (C)WAW
 - (D)WAR
4. 调整指令顺序是解决数据冲突的方法之一. () (2022,2025)

5. 在经典五级流水线 MIPS 处理器中,无法通过数据旁路方式消除数据冲突指令序列是____,____.
(填写两条指令) (2024)



数据冲突下的指令延迟的计算 (8 道题)

1. (本题的前面部分见流水线段数,段长于吞吐量问题的第 1 题 (2016)) 如果将上面的部件形成五级流水线 CPU,分为取指 (F),分析 (D),执行 (E),访存 (M),写回 (W) 五个阶段,每个阶段占一个时钟周期,%edx,%edy,%edz,%edv 为寄存器.以下三条指令按指令流水进行,为了获得最大吞吐率应进行哪些操作?三条指令一共用了多少个时钟周期? (2016)

```
MOV %edx, 100; //将立即数 100 传送到%edx 寄存器
MOV %edy, %edx; //将寄存器%edx 的值传送到%edy 寄存器
ADD %edx, %edy; //%edx = %edx + %edy
```

2. 以下四条指令按指令流水线进行,为了获得最大吞吐率应进行哪些操作?四条指令一共用了多少个时钟周期? (2016)

```
MOV %edx, 100; //将立即数 100 传送到%edx 寄存器
MOV %edy, 200; //将立即数 100 传送到%edy 寄存器
MOV %edz, [%edy]; //将%edy 指向的内存数据传送到%edz
ADD %edv, %edz; //%edv = %edv + %edz
```

3. 指令可以分为 5 个阶段完成,流水线模块延迟为 10ns,流水线寄存器延迟 5ns,执行下面一组指令所需时间至少是多少?分别分析采用旁路技术和不采用旁路技术两种情况. (2018)

```
lw      R1      0(R2)
sub      R3      R1      R4
add      R5      R1      R6
or       R7      R1      R8
add      R9      R1      R10
```



4. 设使用标准 5 级流水线 MIPS CPU,执行下面的程序段,请回答: (2021)

- (1) 指出会发生数据冲突的指令和对应的寄存器.
- (2) 如果 CPU 中没有实现数据转发,则正确执行完程序段至少需要暂停几个时钟?
- (3) 增加数据转发后,至少需要暂停几个时钟?并指出每个数据冲突中被转发的寄存器值应来自于哪个阶段寄存器?

```
LW      R4      0(R5); //将(R5)的值读入到 R4
ADD      R6      R4      R7; //R6=R4+R7
SUB      R8      R5      R9; //R8=R5-R9
```



5. 一个五级流水的 MIPS,L1 Cache 采用指令缓存和数据缓存分别存放,可在一个时钟周期内完成缓存读写 (与寄存器时钟周期同步),现有如下指令已经存入缓存:

指令	指令含义
lw R1 0(R2)	$R1 \leftarrow \text{mem}[R2]$

sub	R4	R1	R5	$R4 \leftarrow R1 - R5$
and	R6	R5	R4	$R6 \leftarrow R1 \& R7$
or	R8	R1	R6	$R8 \leftarrow R1 R6$

- (1) 在无数据旁路的情况下,有多少指令会产生数据冲突? 需要多少个时钟周期完成这些指令?
- (2) 在有数据旁路的情况下,需要多少个时钟周期完成这些指令? 需插入几个气泡? 请说明原因. (2022)



6. 某 MIPS 处理器采用标准 5 级流水线设计,且指令存储器和数据存储器分离.假设一个时钟周期就可以完成 5 级流水线中的任意阶段,同一周期内,寄存器堆写入寄存器的数据即可正确读出,执行下面的指令序列时:



addi \$2, \$3, 4; 3 号寄存器与 4 求和,存入 2 号寄存器
lw \$1,0(\$2); 装入 2 号寄存器指向的内存数据到 1 号寄存器
sub \$4,\$1,\$5; 1 号寄存器与 5 号寄存器求差,存入 4 号寄存器
and \$5,\$1,\$4; 1 号寄存器与 4 号寄存器求与,存入 5 号寄存器

若处理器没有数据旁路,则至少需要插入____个气泡,若有数据旁路,则至少需要插入____个气泡. (2023)

7. 在经典五级流水线 MIPS 处理器中,各个流水段所需要延迟如下(不考虑流水线寄存器的延迟):IF 阶段 130ps,ID 阶段 100ps,无转发的 EXE 阶段 120ps,带全部转发的 EXE 阶段 150ps,仅从 EXE/MEM 转发的 EXE 阶段 140ps,仅从 MEM/WB 转发的 EXE 阶段 130ps,MEM 阶段 120ps,WB 阶段 100ps. (2024)

- (1) 如果实现了全部的转发,则该处理器的时钟周期不短于____ps.
- (2) 如果实现中不使用转发,则该处理器的时钟周期不短于____ps.
- (3) 执行如下的指令序列:



add R5,R2,R1
lw R3,4(R5)
lw R2,0(R2)
or R3,R5,R3
sw R3,0(R5)

实现全部转发,需要____个时钟周期,仅实现从 EXE/MEM 转发,需要____个时钟周期,仅实现从 MEM/WB 转发,需要____个时钟周期. (写出计算过程)

8. 在 RISC-V 典型五级流水线设计,执行一段指令序列



lw x11,16(x10); //从内存中读取一个字符到寄存器 x11 中.
add x13,x11,x12; //将寄存器 x11 和 x12 值相加,结果置于寄存器 x13 中
addi x14,x13,1; //将 x13 寄存器的值与 1 相加,结果置于 x14 中

假定指令每一个执行阶段只需要一个时钟周期,如果仅写回阶段实现数据转发(即在同一时钟周期更新并读取寄存器的值),没有其他数据转发,执行上述指令序列需要____个时钟周期;如果实现全部的数据转发,执行上述指令序列需要____个时钟周期. (2025)

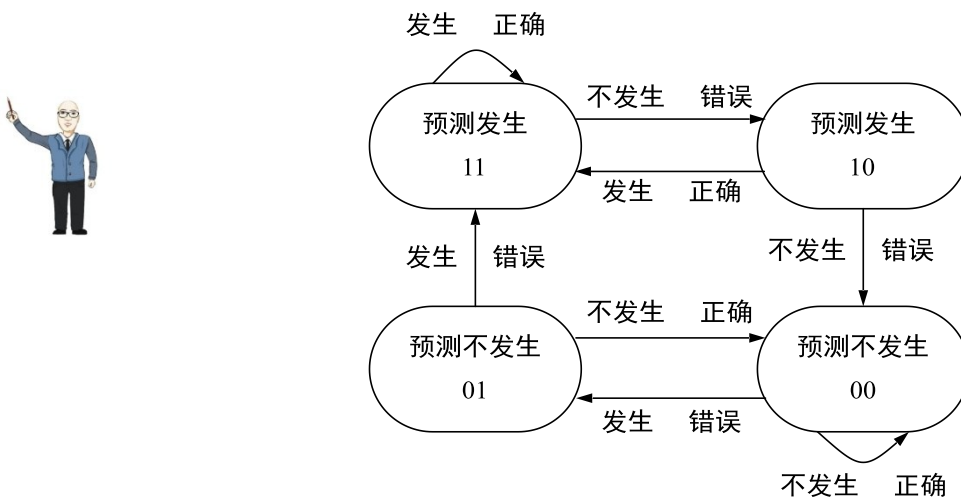
结构冲突的解决方法 (1 道题)

1. MIPS 五级流水线 CPU 中,通过充分设置功能单元可改善结构冲突. () (2019)

控制冲突的解决方法 (1 道题)

1. 分支预测包含了静态预测和动态预测,在下面的 MIPS 程序中,计算不同预测方法下的预测准确率. (2023)

- (1) 采用静态预测,预测转移发生.
(2) 采用 1 位预测位,初始预测转移不发生.
(3) 采用 2 位预测位,初始预测转移不发生.



程序 (\$0 恒为 0):

```
addiu $8,$0,1    $0 + 1 → $8
addiu $9,$0,0    $0 + 1 → $9
```

Loop:

```
add $9,$9,$8      $8 + $9 → $9
addi $8,$8,1      $8 + 1 → $8
addi $1,$8,-1     $8 - 1 → $1
slti $1,$1,100    $1 = { 1, $1 < 100
                     0, $1 ≥ 100
```

```
bne $1,$0,loop    若 $1 ≠ $0, 则 jump 到 loop
nop               空
```

其他流水线冲突问题 (1 道题)

1. 对于 MIPS 流水线 CPU,有如下从 1 加到 100 的程序,下列说法中正确的是 ()

```
main:
    li $t0, 1      #初始化计数器为 1
    li $t1, 0      #初始化总和为 1
loop:
    beq $t0, 101, end_loop #如果计数器等于 101,跳转到 end_loop
    add $t1,$t1,$t0      #将计数器的值加到总和
```

addi \$t1,\$t0,1	#计数器加 1
j loop	#跳回 loop 开始处继续循环
end_loop:	
sw \$t1,sum	#将总和存储到 sum 变量中
#程序结束	
li \$v0,10	#系统调用代码 10 是退出程序
syscall	#执行系统调用

- (A) beq 作为跳转指令,可能会有控制冲突
- (B) 相邻的两轮循环可能因为读写相同的寄存器造成结构冲突
- (C) 分支缓冲优化对这个程序不起作用
- (D) 数据旁路无法完全消除该程序中数据冲突带来的性能损失



具体指令的数据通路 (1 道题)

1. 下列信息传递无关于 lw 指令 (从内存中加载值到寄存器) 功能实现的是 () (2021)
- (A) PC->指令内存 (B) 寄存器堆->ALU (C) ALU->寄存器堆 (D) ALU->数据内存

微程序 (1 道题)

1. 从程序员的角度看,微程序是一种机器语言程序. () (2022)



中断和异常处理 (8 道题)

1. 以下哪个不是 MIPS 响应异常的硬件处理 () (2018)
- (A) 保存 PC (B) 关中断 (C) 保存异常原因 (D) 恢复 PC
2. MIPS 中断中不是由硬件负责的是 () (2020)
- (A) 保存断点 (B) 保存寄存器 (C) 关中断 (D) 保存异常原因
3. 在中断响应过程中,硬件会自动保存 PC 的值,以便中断返回后能继续执行原程序. () (2022)
4. 关于 MIPS 系统的内存缺页异常处理例程的描述正确的是 () (2022)
- (A) 异常处理例程之前,硬件保存通用寄存器内容
- (B) 异常处理例程之后,重新执行发生异常的指令
- (C) 异常处理例程之后,执行发生异常的下一条指令
- (D) 异常处理例程之后,硬件恢复断点
5. 应用程序执行的过程中发生了中断或者异常,假设在指令 A 处发生了中断或者异常,以下描述正确的是 () (2023)
- (A) 如果返回到该应用程序,则开始执行 A 的下一条指令
- (B) 如果返回到该应用程序,则从指令 A 处开始执行 (A 被再一次执行)
- (C) 不会返回到该应用程序执行
- (D) 以上都有可能
6. 尽管 MIPS 只有一套控制寄存器,但是也可以中断嵌套. () (2024)
7. 指令在 () 阶段不会发生异常. (2024)



(A)取指 (B)执行 (C)访存 (D)写回

8. 异常和中断由 CPU 内部或外部事件引起,处理过程相同,对程序执行没有影响. () (2025)

层次存储器 (共 27 道题)

说明:有两道 Cache 的问题,回忆版缺失太多细节,无法修复,故没有展示.

DRAM,SRAM (2 道题)

1. 下列关于静态存储器和动态存储器的描述正确的是 () (2019)

(A)静态存储器使用触发器,需要定期刷新 (B)静态存储器使用电容,不需要定期刷新
(C)动态存储器使用触发器,不需要定期刷新 (D)动态存储器使用电容,需要定期刷新

2. 静态存储器断电后会保存数据. () (2022)

闪存和 FTL (1 道题)

1. 下列关于闪存及 FTL (Flash Translation Layer) 的说法,错误的是 () (2023)

(A)闪存的写入粒度和擦除粒度不同 (B)闪存每单元能编码的比特数越多,寿命越长
(C)FTL 提供逻辑块地址到物理块地址的映射 (D)闪存物理页在更新之前要进行擦除

虚存 (8 道题)

1. 相邻的逻辑地址对应的物理地址不一定相邻. () (2018)

2. 下列选项哪个是对的 () (2019)

(A)虚拟地址空间比物理地址空间大 (B)虚拟地址空间比物理地址空间小
(C)相邻的虚拟地址对应的物理地址也相邻 (D)相邻的虚拟地址对应的物理地址不一定相邻

3. 引入虚拟内存是为了提高访存速度. () (2021)

4. 虚拟存储系统可以加快磁盘的存取速度. () (2023)

5. 虚拟地址到物理地址的映射关系由软件维护. () (2021)

6. 关于 TLB 的功能,以下描述正确的是 () (2021)

(A)TLB 把应用程序访问的数据缓存到 CPU 中
(B)TLB 缺失后,能够在 L1 指令缓存中找到相应内容
(C)TLB 缺失之后会导致应用程序出错
(D)TLB 缓存虚拟地址到物理地址的映射关系

7. 关于层次存储系统的描述正确的是 () (2022)

(A)增加主存容量 (B)实现了时间局部性 (C)实现了空间局部性 (D)提高计算机的性能

8. 某机器按字节编址,虚拟地址占 30 位,物理地址占 28 位,采用一级页表,页面大小 16KB.快表的访问时间是 5ns,Cache 采用直接相连映射,大小 64KB,块大小 4B,Cache 的访问时间 5ns,主存访问时间 40ns.访问次序为:访问快表→访问 cache→访问内存. (2017,部分)

(1) 虚拟页表有脏位 1 位,有效位 1 位,问页表大小?

(2) 系统在进行进程切换时,以下操作是否需要,说明原因:a.清除 cache 有效位.b.清空快表.

Cache 基本原理 (4 道题)

1. 缓存是利用了程序访问的局部性原理. () (2020)
2. 软件管理 Cache 与主存之间的映射关系. () (2023)
3. 下列说法正确的是 () (2018)
(A)缓存越大程序执行速度越快 (B)TLB 也是一种缓存数据和指令的缓存器
(C)指令和数据 Cache 分离可以提高流水线性能 (D)缓存可以提高主存容量
4. 32 位 RISC-V 处理器规格如下,缓存大小 16KB,路数 8,行大小 64B,物理地址 32 位.画出该缓存结构. (2025,部分)



Cache 缺失 (1 道题)

1. 缓存的缺失类型有____,____,____. (任写三个即可) (2019)



Cache 与主存的映射方式 (2 道题)

1. Cache 和主存的几种映射方式为____,____,____. (2018)
2. 在同缓存容量、缓存行大小的情况下,全相联命中率不会小于组相联命中率. () (2021)

Cache-主存系统平均访问时间 (2 道题)

1. 已知 Cache 命中的访问时间为 2ns,Cache 缺失访问时间为 50ns,CPU 执行一段程序时候,CPU 访问内存系统共 10000 次,其中缺失次数为 500 次,那么 CPU 平均访存时间是____. (2024)
2. 某机器按字节编址,Cache 采用直接相连映射,大小 64KB,块大小 4B,Cache 的访问时间 5ns,主存访问时间 40ns.假设 cache 失效访问时间 50ns,命中率为 90%,则此系统的访问某物理地址的平均访问时间为多少? (2017,部分)



标记位,索引位的计算 (3 道题)

1. 某机器按字节编址.虚拟地址 30 位,物理地址 28 位,Cache 采用直接相连映射,大小为 64KB,块大小为 4B,则 28 位的物理地址中的 cache 标志位,索引位,块内地址分别占多少位? (2017,部分)
2. 某按字节编址的计算机层次存储系统,地址位数为 32 位,缓存的大小为 1024 字节,缓存行大小为 8 字节,采用 2 路组相联的方式.索引的位数为____位,标记位为____位. (2021)
3. 32 位 RISC-V 处理器规格如下,缓存大小 16KB,路数 8,行大小 64B,物理地址 32 位.画出该缓存结构,并针对缓存结构给出地址组成包含标志位,索引位和偏移地址. (2025,部分)



Cache 命中率计算 (3 道题)

1. 给出一个指令序列,cache 容量是 1024B,循环 100 次,每次访问 3 和 3+1024 地址的内容,采用直接映射方式的 Cache 命中率为____和采用二路组相联映射方式的 Cache 命中率为____. (2020)
2. 一个 32 位按字节编址的计算机,数据缓存的缓存行的大小为 8B,缓存行有 4 个,缓存状态初始为无效,现执行一个程序,依次访问 0x100,0x110,0x120,0x130,0x100,0x110,0x120,0x130,若为直接映射,命中次数为____,若为全相联,采用 LRU 替换策略,命中次数为____. (2022)

3. 32 位 RISC-V 处理器规格如下,缓存大小 16KB,路数 8,行大小 64B,物理地址 32 位.对二维数组 `int A[8][1024]`,执行代码片段:

```
for(i=1;i<4;i++){
    for(j=0;j<1024;j++){
        A[i][j]=A[0][j];
    }
}
```



假设 `i` 和 `j` 保存在寄存器中,数组的首字母为 0,数据缓存初始为空,采用缺失读取和 LRU 替换策略该代码共产生多少数据缺失,缺失率多少? (2025,部分)

虚拟 Cache (1 道题)

1. 某机器采用按字节寻址的方式.30 位虚拟地址,28 位物理地址,一级页表,页面大小 16KB.快表的访问时间是 5ns,Cache 采用直接相连映射,大小 64KB,块大小 4B,访问时间 5ns,主存访问时间 40ns.访问次序为:访问快表→访问 cache→访问内存.注意到快表访问和 cache 访问时间相同,能否修改访问方式,使 cache 和页表一同访问? 说明原因. (2017,部分)

IO 和总线 (共 19 道题)

RAID 的容错能力 (3 道题)

1. 采用 () 的磁盘不具备容错能力 () (2019)
(A)RAID0 (B)RAID1 (C)RAID5 (D)RAID6
2. 如果采用 RAID6,则在两个磁盘损坏的情况下,也可以修复. () (2018)
3. RAID5 比 RAID4 容错性更好. () (2025)

RAID 的磁盘大小计算 (1 道题)

1. 存储 100MB 的数据,RAID1 的磁盘的大小为____,RAID5 的磁盘的大小为____. (2020)

RAID 的读写速度 (1 道题)

1. RAID1 的磁盘比不使用 RAID 的磁盘读写更快. () (2024)



RAID5 数据恢复 (2 道题)

1. 一个 4+1 的 RAID5 磁盘组织,同一个地址在前四个磁盘中的数据分别为 0x11,0x22,0x33,0x44,此时第五个磁盘出错,新磁盘替换后,新磁盘中该地址上的数据初始化为____. (2022)
2. RAID5 磁盘同一个条带中数据块对应位置的二进制数据分别为 1011,0110,1001,1110,那么校验块对应位置的数据为____. (2024)

DMA (4 道题)

1. DMA 使用总线的方式有____和____. (2017)

2. 相对于程序直接控制方式,DMA 可以提高硬盘到内存的装入速度. () (2018)
3. DMA 设备不可独占内存总线周期. () (2021)
4. 下关于通过 DMA 方式进行数据传输的描述,错误的是 () (2023)
 - (A)DMA 用于数据快速传输,可用来代替中断方式
 - (B)DMA 控制器与 CPU 交替使用总线
 - (C)DMA 可以直接访问主存
 - (D)数据传输过程由 DMA 自行控制



磁盘 (2 道题)

1. 对于机械硬盘,读取顺序存储的文件比读取随机存储的文件速度快. () (2019)
2. 磁盘转速为 10000 转每分钟,传输速度为 20MB/s,平均寻道时间为 7ms,控制器延迟为 0.3ms,读取 4KB 扇区数据大约需要多少时间? (2024)
 - (A)10.1ms
 - (B)10.5ms
 - (C)11.5ms
 - (D)13.5ms

IO 端口的编址 (1 道题)

1. IO 端口和主存可以统一编址. () (2023)



显存带宽 (1 道题)

1. 一台计算机显示器的分辨率为 800*600,使用 RGB 颜色,每个颜色使用 1 个字节表达,帧率为 50Hz,显示器的总带宽的 80%用于刷新屏幕,则需要的显存带宽至少为____. (2020)

总线带宽 (2 道题)

1. 一个 32 位的总线系统,时钟频率 200MHZ,2 个时钟周期传完一次,总线的带宽是 () (2022)
 - (A)200MB
 - (B)400MB
 - (C)600MB
 - (D)800MB
2. 某系统总线可在一个总线周期内并行传输 16B 的数据,一个总线周期占用 3 个时钟周期,总线的时钟频率为 66MHz,则总线带宽是____. (2023)



总线仲裁 (1 道题)

1. 菊链式仲裁比集中平行仲裁公平性更好. (2025)

同步总线,异步总线,单总线,双总线 (1 道题)

1. 下列关于总线的说法中,正确的是 () (2019)
 - (A)并行总线速度大于串行总线
 - (B)异步总线速度大于同步总线
 - (C)单总线速度大于双总线
 - (D)以上说法均错误

操作系统

说明:2016,2017 年的 OS 多选题是至少有一个选项正确的那种多选,2016-2025 年只有 16,17 出过多选.

绪论 (共 17 道题)

中断、异常 (7 道题)



1. 由 x86 硬件完成的是 () (多选题) (2017)
(A)获取中断源 (B)形成中断入口地址 (C)Eax 寄存器保存 (D)新威考研
2. ucore 的时钟中断设为 10ms 触发一次,所以 ucore 不能实现小于 10ms 的周期定时间隔.() (2019)
3. 操作系统可通过对特定标志寄存器的设置来屏蔽时钟中断 () (2021)
4. 外部设备产生中断后,操作系统一定会马上响应中断. () (2022)
5. 若应用程序执行中外设产生中断,则操作系统接管 CPU 控制权并完成中断处理. () (2023)
6. GDB 调试不需要操作系统的服务来跟踪和中断被调试应用程序的执行 () (2023)
7. RISC-V 处理器在 U 态 (用户态) 下执行 ecall 指令会陷入到 S 态 (内核态) 中下列关于硬件会自动完成的操作中错误的是 () (2023)
(A)保存进程的页面基址 (B)保存 ecall 执行指令的所在地址
(C)保存陷入原因“在 U 态执行 ecall 指令” (D)转跳到异常处理例程的入口地址

系统调用 (4 道题)

1. 给出系统调用的四个分类,例如 xx 类完成 xx 的创建、撤销和退出. (2020)
2. 给出 getpid 系统调用的执行过程 (2020) _____
3. 以下系统调用允许的最大参数个数为_____个,其中 num 是_____,edx 和 ecx 存储的分别是第_____和第_____个参数,系统调用的返回值存放在_____. (2020)



```
syscall(int num, ...) {
    va_list ap;
    va_start(ap, num);
    uint32_t a[MAX_ARGS];
    int i, ret;
    for (i = 0; i < MAX_ARGS; i++) {
        a[i] = va_arg(ap, uint32_t);
    } va_end(ap);

    asm volatile (
        "int $1;"
        : "=a" (ret)
        : "i" (T_SYSCALL),
          "a" (num),
          "d" (a[0]),
          "c" (a[1]),
          "b" (a[2]),
          "D" (a[3]),
          "S" (a[4])
        : "cc", "memory");
    return ret;
}
```

4. 应用程序执行系统调用可能会引起进程上下文切换. () (2024)

操作系统内核,微内核 (4 道题)

1. 下列组件不属于操作系统内核的是 () (2023)
(A)JavaScript 解释器 (B)文件系统 (C)中断处理例程 (D)进程调度器
2. 微内核操作系统的设备驱动程序大多运行在内核态. () (2024)

3. windows 进程运行在用户态可直接访问所有内存区域. () (2025)
4. 微内核架构设计可提高系统的模块化程度和安全性. () (2025)

Ucore 和其他问题 (2 道题)

说明:连续五年没考过 ucore 或者 rcore 实验了,不再展示早年考过的 ucore 实验题.

1. $CPL \leq DPL[\text{门}]$ 和 $CPL > DPL[\text{段}]$ ____ 表示请求时可以和门特权级相同, ____ 表示请求时应低于段的特权级. (2019)
2. qemu-system-riscv64 可以模拟 64 位 RISC-V 处理机并让操作系统在其上运行 () (2023)



内存管理 (共 28 道题)

首次匹配,最佳匹配,最差匹配 (2 道题)

1. 以下哪种算法会产生很多不必要的小碎片的分区 () (多选题) (2017)
(A)最佳匹配算法 (B)首次适应算法 (C)最坏匹配算法
2. 能够有效避免产生小碎片的算法是 () (2017)
(A)最佳匹配算法 (B)首次适应算法 (C)最坏匹配算法

伙伴系统 (1 道题)

1. Buddy 算法中,释放一个空间后可以根据其起始地址和大小与相邻空闲空间合并. () (2018)



Uncached 的概念 (1 道题)

1. 下列说法中正确的是 () (多选题) (2016)
(A)失忆 (公众号: 新威考研) (B)回忆失败 (www.xinweikaoyan.com)
(C)有些数据 cache 并不能存放 (D)多核处理器可以有共享的一个高速缓存

分页内存管理的基本概念 (8 道题)

1. x86 计算机中,CR3 应该保存页目录表的起始____. (2019)
2. 分页内存管理下,每个进程都有自己的页表. () (2020)
3. 通过页表机制可以实现进程间共享内存 () (2022)
4. 在没有 MMU 的 CPU 上 Linux 内核可正常运行并具备基于页表机制的虚拟存储管理能力 () (2023)
5. 请求分页存储管理需要 CPU 硬件的支持. () (2024)
6. 操作系统的分页机制不能同时避免内碎片和外碎片. () (2025)
7. 通过页表机制和特权级机制可防止用户态进程之间的非法访问. () (2025)
8. 切换进程时,进程切换不仅会保存和恢复寄存器,还会保存和恢复页表. () (2025)



求页面大小（1 道题）

1. 一台计算机虚拟地址空间为 8KB,物理地址空间为 4KB,采用二级页表的方式组织内存,页表项是 2B,页目录项 1B,一级页表和二级页表大小均为 32B,则页面大小是多大? (2017)



多级页表（3 道题）

1. 采用二级页表的 x86-32 虚拟存储系统,其一级页表可以不在内存中. () (2019)
2. 当进程用完所有虚拟地址空间后,多级页表比一级页表所占的内存空间大 () (2022)
3. 一个进程的页表中,页目录项和页表项不同时存在. () (2024)

具体的逻辑地址到物理地址的转换（2 道题）

1. 大题 (3+3+4) (2023)

虚拟内存管理是操作系统中非常重要的组成部分,请回答以下问题

(1)请阐述操作系统中虚拟内存管理的基本功能

(2)请说明 mmap()和 munmap()系统调用的含义及基本实现机制

(3)某 32 位处理机中虚拟页面存储方式的页面大小为 4KB,采用 32 位虚拟地址,34 位物理地址,二级页表,每个页表包含 1024 项,页表项大小为 4B,请回答以下问题

A.请画图并简要描述虚拟地址转到物理地址的地址转换过程

B.请在图中标注和描述页表项各部分内容的含义和相关控制状态寄存器的作用



2. 有一台假想计算机,虚拟页和物理页帧大小均为 16Bytes,支持 256Bytes 的虚拟地址空间,有 256Bytes 的物理内存;采用二级页表,一个页目录项大小为 1Byte,一个页表项大小为 1Byte,1 个页目录大小为 4Bytes,页目录基址寄存器保存了页目录表的物理页帧号,占 1Byte. (2024)

PTE 和 PDE 格式 (8bits) :

Frame Number	I	R	W	X
4bits	1	1	1	1

其中,I,R,W,X 都是标志位,分别是存在位,可读位,可写位,执行位,Frame Number 是物理页帧号.

页目录基址寄存器格式为:

Frame Number	Reserved
4bits	4bits



页目录基址寄存器的内容为 0x00

256Bytes 的整个物理内存内容如下: (其中数值都是十六进制)

0x00	0C	1C	2C	3C	00	00	00	00	00	00	00	00	00	00	00	00
0x10	4C	5C	6C	7C	00	00	00	00	00	00	00	00	00	00	00	00
0x20	8C	9C	AC	BC	00	00	00	00	00	00	00	00	00	00	00	00
0x30	CC	DC	EC	FC	00	00	00	00	00	00	00	00	00	00	00	00
0x40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

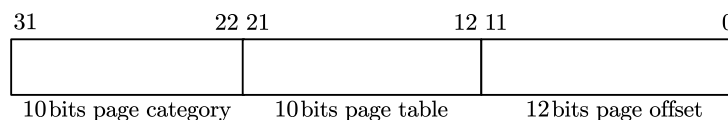
0x60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xa0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xb0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xc0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xd0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xe0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0xf0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

- (1) 在这台假象计算机的虚拟地址中,页目录号,页表号和页内偏移各占多少位?请简明解释求解过程.
- (2) 在这台假想的计算机上执行访存指令时,虚拟地址 0x32 经地址转换后得到的物理地址为多少?内容是什么? 请简明解释过程.
- (3) 已知这台假想的计算机上执行访存指令时,虚拟地址 0x12 经过地址转换后得到的物理地址为 0x12, 请回答对应页目录号和内容,对应页表号和内容分别是什么? 请简明解释过程.



PAE 技术 (1 道题)

1. Intel-x86-32CPU 使用分页管理,每页 4KB,逻辑地址格式如图所示,这种 CPU 最多支持 4GB 内存.为了使系统能够使用 64GB 内存,使用物理内存扩展技术,使物理地址长度变为 64 位,页面大小仍为 4KB.试设计逻辑地址格式,使其可以在支持物理内存扩展技术的 x86-32CPU 上运行. (2018)



反置页表 (1 道题)

1. 反置页表的每一个页表项中存在①____、②____、③____和 hash 冲突页表项链表指针.在反置页表地址转换的过程中,hash 函数的输入是④____和⑤____,输出用作⑥____.如果页表项中的内容和 hash 函数的输入不一致,则会出现⑦____冲突.
A.进程控制块 B.文件标识符 C.hash 冲突 D.标志位 E.物理页号 F.虚拟页号 G.进程 ID (2022)

虚拟地址空间和物理地址空间大小比较 (2 道题)

1. 虚拟存储系统允许进程的虚拟地址空间大于实际物理内存大小. () (2024)
2. 虚拟内存技术限制程序使用虚拟地址空间不能小于物理内存空间实际的大小. () (2025)



页面置换算法 (6 道题)

1. 以下会发生 Belady 异常的是 () (多选题) (2017)
(A)FIFO (B)LRU (C)CLOCK (D)LFU

(E)改进版 CLOCK

2. OPT、FIFO、LRU、CLOCK、LFU 这些页面置换算法,哪些可能出现 Belady 现象? 可能的举例,不可能的证明 (2018)
3. FIFO 算法存在 Belady 现象. () (2020)
4. 时钟页面置换算法,能通过读取和修改页表项中的访问位来实现. () (2021)
5. 最优页面置换算法可得到最少的缺页率,因此在操作系统中优先采用 () (2021)
6. 最优页面置换算法 (OPT) 属于栈式算法 (Stack),存在 Belady 异常现象 () (2022)

进程管理 (共 54 道题)

程序和进程的关系 (2 道题)

1. 关于程序和进程的关系,以下描述正确的是 () (多选题) (2016)
(A)进程是动态的,程序是静态的 (B)进程是暂时的,程序是永久的
(C)进程的组成包括程序,数据和进程控制块 (D)同一个程序的多次运行对应相同的进程
2. 进程是应用程序的一次执行过程. () (2023)

进程切换,线程切换 (4 道题)



1. 8 位 CPU 不能进行线程切换. () (2016)
2. 切换进程上下文时,CPU 硬件会自动保存当前程序计数器 (PC),无需操作系统干涉. () (2025)
3. 下面有关进程切换的描述,错误的是 () (2023)
(A)进程切换是来自两个不同进程在内核态中控制流之间的切换
(B)进程切换的过程可以在内核态执行也可以在用户态执行
(C)外设产生的中断不一定会导致进程切换
(D)进程切换的过程涉及到处理器中寄存器的保存与恢复
4. 进程切换可以发生在用户态,也可以发生在内核态. () (2024)



进程状态转换 (3 道题)

1. 如果用户强制使用任务管理器 kill 一个进程,那么即使它处于就绪状态/阻塞状态,操作系统也要把它变成运行状态 () (2018)
2. 运行态不能转换到就绪态 () (2021)
3. 如果操作系统中有 M 个进程,则就绪队列中最多有 M-1 个进程. () (2021)

用户态线程问题 (1 道题)

1. 用户态线程可直接执行特权级指令. () (2025)



其他进程和线程相关问题 (3 道题)

1. 只有一个 main 函数的进程没有线程. () (2019)
2. PCB 是操作系统内核为进程设置的管理数据结构,每个进程有且仅有一个 () (2021)
3. 在支持多线程的操作系统中,可以通过进程控制块访问该进程的线程执行状态. () (2024)

fork()系统调用 (2 道题)

1. 采用 copy on write 机制的 fork()系统调用不会复制进程的页目录表. () (2018)
2. 下列关于 fork()系统调用实现的说法错误的是 () (2023)
 - (A)父子进程在返回到用户态的瞬间,都处于刚从系统调用返回的状态,但是二者的返回值不同
 - (B)fork()系统调用会创建一个不同于父进程的子进程的内存栈
 - (C)fork()系统调用会给予子进程创建一个进程块,其中 TrapContext 部分的内容和父进程相同
 - (D)父进程创建的子进程后,二者的进程 ID 一定不同



exec()系统调用 (1 道题)

1. exec()系统调用会改变以下哪些参数 () (多选题) (2017)
 - (A)进程 ID
 - (B)父进程 ID
 - (C)打开文件指针
 - (D)公众号:新威考研

僵尸进程,孤儿进程 (4 道题)

1. 若子进程执行 exit()退出,若未检测到父进程执行 wait(),则子进程进入____状态. (2018)
2. 父进程先退出,未退出的子进程被称为____.子进程退出时,其父进程还没 wait,此时子进程被称为____. (2019)
3. 父进程退出时,如果有子进程,子进程将会变成僵尸进程. () (2021,2024)



进程调度 (4 道题)

1. 进程调度算法中,高响应比优先调度中的“响应比”的分母是程序的(执行时间),分子是(执行时间+等待时间) (2018)
2. stride 调度算法中,假设用八位无符号数来表示 stride,进程 A 的 stride 有可能会溢出,但是当步长 $s \leq 127$ 时,只要利用一个小小的技巧就仍可以正确调度.假设有进程 A 和 B, a 为 A 的 stride, b 为 B 的 stride,当 $\text{sign}(a-b) \leq 0$ 时说明 A 正在运行,此时 $a+s \leq 256$;当 $\text{sign}(a+s-b) \leq 0$ 时,轮到 B 运行. (备选的选项: $\leq, \geq$) (2019)
3. SJF 调度算法可能会导致长作业饥饿. () (2020)
4. 最短作业优先调度算法 (SJF) 可以保证作业的平均等待时间最短. () (2024)

禁用中断 (3 道题)

1. 在单核 CPU 中,可以通过屏蔽/使能硬件中断来实现进程的同步互斥 () (2022,2024)

2. 在多处理器场景下控制中断的打开与关闭,可以保证只有一个线程进入临界区. () (2023)

软件同步 (1 道题)

1. 以下是一个软件同步算法的伪代码,要求支持线程 $p[0]$ 和 $p[1]$ 的临界区访问

(1) 解释该代码的原理

(2) 完成填空,配合必要的注释 (2022)

variables:

wants_to_enter: array of two booleans

turn: integer

wants_to_enter[0] ← false

wants_to_enter[1] ← false

turn ← 0 // or 1

线程 $p[i]$ 的算法伪代码 (i 的值取 0 或 1):

wants_to_enter[i] ← ①

while(wants_to_enter[②]) {

if(turn ③ i) {

wants_to_enter[i] ← ④

while(turn ⑤ i) {

// busy wait

}

wants_to_enter[i] ← ⑥

}

}

临界区代码:

critical_section():

turn ← ⑦

wants_to_enter[i] ← ⑧

// remained section

自旋锁 (1 道题)

1. 使用自旋锁不能保证进程按先来先服务的顺序使用 cpu 资源. () (2018)

信号量 (3 道题)

1. 信号量的四个填空: (2019)

```
class Semaphore{
    int sem;
    WaitQueue q;
    Semaphore::P(){
```



```

    ____;
    if(____){
        Add this thread t to q;
        block(t);
    }
}
Semaphore::V(){
    ____;
    if(____){
        remove a thread t from q;
        wakeup(t);
    }
}
}

```

2. 信号量可用于实现同步 (synchronization) 比如将信号量的初始值设为 0, 若线程 A 对此信号量执行一个 P 操作那么该线程应当被阻塞睡眠 () (2023)

3. 信号量 (2025)

两个线程 P1 和 P2 互斥使用一个包含 N 个单元的缓冲区 buffer 进行数据传递. P1 每次用 produce() 生成一个随机整数 item, 并用 put() 把 item 放入缓冲区某一空单元中 P2 每次用 buffer_has_item_bigger_than_zero() 判断是否有正整数 item, 如果有, 则用 get() 从该缓冲区中取一个正整数 item. 如果没有, 则进一步判断缓冲区是否满. 满时用 empty_buffer() 取出所有 item.

下面是两个线程同步与互斥活动的伪代码, 请尝试在 10 个填空位填写信号量相关的 PV 操作, 使其满足上述要求, 如无法满足, 请说明理由.

//初始化信号量

```

semaphore mutex = 1;      //用于保护缓冲区的互斥访问
semaphore empty = ( );    //填空 1, 记录缓冲区中空单元的数量
semaphore positive = 0;   //记录缓冲区正整数的数量

```

//P 操作

```

P(semaphore s){
    s.value--;
    if(s.value < 0){
        wait();    //将调用线程放入等待队列
    }    //end if
}    //end P

```

//V 操作

```

V(semaphore s){
    s.value++;
    if(s.value >= 0){
        wake up();    //从等待队列中唤醒一个线程
    }    //end if
}    //end V

```

//缓冲区



```
int buffer[N];
```

//线程 P1-生产者

```
thread P1 {
    while(true){
        int item = produce();    //生成一个随机整数
        ();    //填空 2
        ();    //填空 3
        put(item,buffer);    //将 item 放入缓冲区 buffer 中
        ();    //填空 4
        if(item > 0){
            ();    //填空 5
        }    //end if
    }    //end while
}    //end thread
```



//线程 P2-消费者

```
thread P1 {
    while(true){
        ();    //填空 6
        if(buffer_has_item_bigger_than_zero()){    //是否有正整数 item
            ();    //填空 7
            get(buffer);    //从缓冲区取一个正整数 item
            ();    //填空 8
        } else if(buffer_is_full()){    //缓冲区是否满了
            empty_buffer();    //取出所有 item
            int i = 0;
            while (i < M){
                ();    //填空 9
                i++;
            }    //end while
        }
        ();    //填空 10
    }    //end while
}    //end thread
```

管程（7 道题）

- 关于线程和管程,下面说法错误的是 (回忆版题目不完整,新威考研强行补充选项) (2017)
 - 管程提供了互斥机制,确保一次只有一个线程可以执行管程中的代码
 - 线程在管程中等待条件变量时,会释放管程的锁,以便其他线程进入管程
 - 管程中的条件变量用于线程间的同步,允许线程等待特定条件成立
 - 多个线程可以同时进入管程,但只能访问不同的条件变量
- ____支持线程暂时放弃互斥资源访问权,等待信号. (2018)
- ____提供了一个执行环境,其中只能同时执行一个线程. () (2018)

4. 管程好比一个黑箱子,程序员往里面扔函数,同一时间只有一个函数在执行 () (2018)
5. 管程和信号量在功能上等价 () (2018)
6. 管程将资源抽象成条件变量,通过对条件变量的值的增减来控制进程的访问. () (2018)
7. 管程可以用信号量来实现. () (2022)

死锁问题的必要条件 (2 道题)

1. 死锁必要条件:互斥访问,占有并等待,非剥夺,循环等待. () (2016)
2. 死锁的充分必要条件是互斥,持有并等待. () (2020)



哲学家进餐问题 (1 道题)

1. 学家进餐问题 (2018)

```
#define N 5
semaphore fork[5];
void philosopher(int i){
    while(TRUE){
        think();
        P(fork[i]);
        P(fork[(i+1) % N]);
        eat();
        V(fork[i]);
        V(fork[(i+1) % N]);
    }
}
```



- (1) 上述算法会不会死锁,如果会请举例.
- (2) 上述算法是否允许两个哲学家同时进餐,若可以请举例

银行家算法 (4 道题)

1. 银行家算法中,不安全状态即死锁状态. () (2019)
2. 如果操作系统采用银行家算法,则操作系统处于不安全状态时可能会发生死锁 () (2021)
3. 银行家算法可以避免死锁. () (2024)
4. 银行家算法可以预防死锁. () (2025)

其他死锁相关问题 (3 道题)

1. 101 个共享资源,m 个进程,每个进程需要 5 个资源,如果不发生死锁,则 m 最大值为 () (2020)
(A)24 (B)25 (C)100 (D)101
2. 信号量机制可以解决进程死锁的问题. () (2020)
3. 先来先服务进程调度算法 (FCFS) 可以解决进程间死锁的问题. () (2022)

进程通信 (2 道题)

1. 简答 (2021)

对于信号、信号量、管道、消息队列,以下情况用哪个方式更为合适?

A) 一个进程 kill 掉另一个进程.

B) linux 命令行下运行命令 `cat txtfile.txt | grep exam`

C) A 进程完成 2MB 结构化数据的输入,再传给 B,并基于此数据的一个关键域进行排序

2. 在 shell 中输入“ls | more”,shell 会建立两个子进程,并在这两个进程之间建立管道(pipe)() (2022)

多处理器 (2 道题)

1. 下列关于操作系统的描述中错误的是 () (2023)

(A)支持多核处理器的操作系统中允许多个执行程序并行执行

(B)支持多核处理器的操作系统中允许多个执行程序并发执行

(C)仅支持单核处理器的操作系统中允许多个执行程序并行执行

(D)仅支持单核处理器的操作系统中允许多个执行程序并发执行



2. 在多核处理器中,采用用户线程实现多线程时,同一进程的多个线程可以并行执行 () (2023)

优先级反置 (1 道题)

1. 优先级反置指的是______抢占了______的资源,优先级继承在只有在高优先级进程______时,低优先级进程能动态改变优先级. (2018)

文件管理 (共 11 道题)

说明:有 1 道题失忆太多,无法修复,不再展示

文件分配 (1 道题)

1. 一个文件系统采用索引结点方式存储文件,一个索引结点包括两个直接文件指针,一个一级间接文件指针表,一个存储块的大小为 8KB,一个指针大小为 4B,则该文件系统能存放的最大文件是多大? 用 TB+GB+MB+KB+B 表示 (2016)



文件控制块,inode (3 道题)

1. UNIX 索引结构存放的位置是 () (2020)

(A)超级块

(B)索引节点

(C)目录项

(D)空闲块

2. 文件控制块除了包含了元数据信息,也可以包含文件数据内容. () (2024)

3. 基于 inode 文件系统中,inode 包含了文件元数据信息,但不包括文件名. () (2025)

硬链接和软连接 (2 道题)

1. A 是引用计数为 1 的文件,创建 A 的硬链接 B,再创建 A 的软链接 C,再创建 B 的硬链接 D,则此时

B 的引用计数为____,D 的引用计数为____. (2019)

2. file1 引用计数为 2,建立 file1 的一个软链接 file2,建立 file1 的硬链接 file3,建立 file3 的硬链接 file4,此时 file3 和 file4 的引用计数分别为 () (2020)
- (A)4,4 (B)3,4 (C)4,3 (D)2,4

其他文件相关问题 (5 道题)

1. 在一个硬盘中可以存在多种不同的文件系统 () (2022)
2. 操作系统中的文件系统必须具备打开文件,关闭文件,读文件,写文件的功能 () (2023)
3. 应用程序每次调用 open()函数打开文件都要通过系统调用向操作系统提出请求 () (2023)
4. 删除一个文件后,该文件的所在的当前目录将改变. () (2020)
5. PCB 中的当前工作目录可以加速文件的查找. () (2020)



设备管理 (共 4 道题)

RAID (1 道题)

1. 以下哪种磁盘阵列存取速度快 () (多选题) (2017)
- (A)RAID0 (B)RAID1 (C)RAID4 (D)RAID5

延迟写 (1 道题)

1. 延迟写操作可以减少对磁盘的访问次数. () (2020)

SSTF 对 SSD 无效 (1 道题)

1. 最短寻道时间算法在 SSD 存储设备中无效. () (2020)

Linux 把外设表示成文件 (1 道题)

1. linux 操作系统中把外设表示成文件,让应用程序以文件操作的形式来访问外设 () (2022)



计算机网络

绪论 (共 7 道题)

协议,标准 (2 道题)

1. 制定 RFC 的标准化组织是 IETF. (2025)
2. 网络协议由语法、语义和时序三个要素组成. (2025)



面向连接,无连接 (1 道题)

1. 面向连接的网络服务是可靠服务. (2025)

电路交换,报文交换,分组交换 (1 道题)

1. 关于电路交换和分组交换,下列说法中正确的是 () (2019) (原本选项失忆,新威考研强行修复)
(A)电路交换在通信过程中独占链路资源,分组交换则通过“共享资源+存储转发”传输数据
(B)电路交换的端到端时延抖动大,分组交换的端到端时延固定且可预测
(C)语音通话等实时业务更适合用分组交换,文件下载等非实时业务更适合用电路交换
(D)分组交换需要先建立专用通信链路(如拨号),电路交换无需建立连接即可直接传输数据

发送时延,传播时延 (1 道题)

1. 主机甲通过一个路由器(存储转发方式)与主机乙互连,两段链路的数据传输速率均为 10Mbps,主机甲采用报文交换向主机乙发生 1 个大小为 8Mb ($1\text{M}=10^6$) 的报文。若忽略链路传播延迟,完成该报文传输所需的时间为 () (题目没会回忆起来,改编一道 408 真题) (2019)
(A)1.5s (B)1.6s (C)2.1s (D)3.1s

OSI 和 TCP/IP 参考模型 (2 道题)

1. 目前应用最广泛的计算机网络体系参考模型是 ISO/OSI. () (2025)
2. 关于 TCP/IP 与 OSI 说法正确的是 () (2018)
(A)TCP/IP 模型分为五层
(B)OSI 模型从上到下依次是应用层,会话层,表示层,传输层,网络层,数据链路层,物理层
(C)TCP/IP 模型从上到下依次是应用层,网络层,数据链路层,物理层
(D)TCP/IP 模型适用场合比 OSI 模型更广



物理层 (共 9 道题)

香农定理,奈奎斯特定理 (2 道题)

1. 适用于奈奎斯特定理的是 () (2018)

I. 光纤

II. 同轴电缆

III. 红外线

(A) I 和 II

(B) II 和 III

(C) I 和 III

(D) I、II、III

2. 香农定理适用于各种传输介质. () (2025)

蜂窝网络 (1 道题)

1. 蜂窝移动网络六边形, 频率 840Hz, 每个单元可使用最大频率个数 () (2019)

(A) 280Hz

(B) 140Hz

(C) 560Hz

(D) 240Hz

物理层设备 (2 道题)

1. 以下设备只工作在物理层的是 () (2020)

(A) 交换机

(B) 中继器

(C) 路由器

(D) 网桥

2. 中继器是物理层中的网络互连设备 () (2023)



数字通信, 模拟通信 (2 道题)

1. 数字数据在模拟信号中传播需要的设备是 () (2020)

(A) 调制解调器

(B) 编码解码器

(C) 公众号

(D) 新威考研

2. 基带传输用数字信号传输数字数据. (2025)

光网络, 传输介质 (1 道题)

1. 光网络只能通过导向型介质传播 () (2023)



信道复用技术 (1 道题)

1. 下列不属于静态信道复用技术的是 () (2023)

(A) 频分复用

(B) 时分复用

(C) 码分复用

(D) 统计复用

数据链路层 (共 13 道题)

数据链路层概述和功能 (2 道题)

1. ISO/OSI 模型中的____层会在处理协议元数据的时候在数据首尾添加信息. (2022)

2. 数据链路层使用的传输单位是 () (2018)

(A) 比特

(B) 报文

(C) 帧

(D) 分组

组帧 (1 道题)

1. 数据链路层成帧的方法不包括 () 法 (2022)

(A) 偏移量

(B) 物理层编码违例

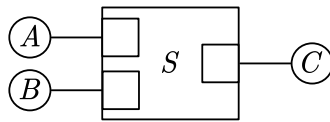
(C) 带位填充的定界符

(D) 字节计数

信道利用率,最大数据传输速率 (4 道题)

1. 在带宽为 128kbps 的无错信道上,某太空站向地球单向发送 512 字节的数据帧,端到端的传播时延为 300ms,确认帧长度忽略不计,接收窗口足够大,问发送窗口分别为 1,15,27 时,吞吐率是多少?若要使信道利用率达到最大,则帧序号至少需要多少位? (2017)
2. 两地相距 3000 公里,传播速度为 $6\mu\text{s}/\text{公里}$,传输 64 字节的帧,采用 GBN 协议,带宽为 1.536Mbps,则若要最大限度发挥网络带宽,至少需要 () 位比特的序号 (2018)
(A)4 (B)5 (C)7 (D)8
3. 下列哪些情况会使得停等协议通信线路利用率最低 () (2019)
(A)源和目的之间距离很近,发送速率快 (B)源和目的之间距离很近,发送速率慢
(C)源和目的之间距离很远,发送速率快 (D)源和目的之间距离很远,发送速率慢
4. 数据链路层计算题 (2022) (回忆版太模糊,被新威考研强势修复)

如图,网络设备 S 和主机 A、B、C 相连,S 的每个接口的最大吞吐量和全部链路的信道带宽都是 1000 帧每秒,每帧承载有效载荷为 1KB,数据帧处理只考虑转发时间,每条链路一样长,传播时延都是 2ms,链路上无误码,ACK 帧不计算处理和转发的时间,每个主机采用滑动窗口控制拥塞,发送、接收、拥塞窗口大小单位为帧。



- 1) 若 A 发送窗口大小为 1,A 向 C 稳定发送足够多的数据,当 S 为集线器或存储转发交换机时,与 A 相连的链路效率分别为多少?
- 2) 若 S 为全双工存储转发式交换机,A 向 C 发送 30KB 数据,若要做到最快且无拥塞,A 发送窗口最小为多少?传输 30KB 的数据最短需要多长时间?
- 3) 若 S 为全双工存储转发式交换机,S 的接口都有 4 帧缓存,C 的接收窗口足够大,A 和 B 都向 C 发送数据,若要不产生拥塞丢失,并且公平使用两条链路,则 A 和 B 的拥塞窗口最大为多少?



GBN、SR (4 道题)

1. 在选择重传协议中,序号为 0~13,若发送窗口大小为 7,为了使传输不出错,接受窗口大小最大为 () (2018)
(A)4 (B)5 (C)7 (D)8
2. 在选择重传协议中,当前发送方的发送窗口为[1,2,3,4],收到了一个接收方的否定确认帧,则可能的情况是 () (2020)
(A)一定是 1 (B)一定是 2 (C)一定是 3 (D)可能是 1,2,3,4
3. 通信双方使用滑动窗口退后 N 帧重传协议(GBN),采用累积确认,发送方已经发送 3 比特序号位的序号为 0~7 的帧,若发送方只收到 0,2,3 号帧的确认,计时器超时会导致发送方重传 4 个帧. () (2025)
4. A 向 B 传输 8 个长度为 Lbits 的分组,每组传输结束后启动定时器,链路速率 Cbps,单向传播延时 $T_s, T \gg L/C$,假设 A 的发送窗口足够大,采用超时重传机制,超时时间为 2 倍的往返传播时延,分别采用 GBN 协议、选择性重传协议和带否定性确认机制的选择重传协议 (否定性确认分组长度可

忽略) 实现可靠传输, 假定分组间无间隙, 若第 2 组 (从 1 开始编号) 出现误码, 之后无错, 选择性重传协议和带否定性确认机制的选择重传协议的接收窗口足够大, 则采用以上三个机制, 从发送方开始发送第一个分组到接收方正确接收第八个分组分别需要多少时间? (2021)

CRC (1 道题)

1. 对于 CRC 产生式 $G(x)=x^2+x+1$, 下面信息正确的是 () (2021)
- (A) 11101010 (B) 11001111 (C) 10001011 (D) 11100011

海明码 (1 道题)

1. 可以纠正 n 位错误的海明码的最短码距是 _____. (2022)



MAC 子层 (共 13 道题)

以太网, CSMA/CD (4 道题)

1. 假设以太网中最短帧长为 1000bit, 最远两点距离 100m, 传播速度为 2×10^8 m/s, 最大发送速率为 () (2019)
- (A) 1Gbps (B) 2Gbps (C) 100Mbps (D) 200Mbps
2. 一个 CSMA/CD 协议的局域网中, 介质为电缆, 链路速率为 1Gbps, 信号传播速率为 200000km/s, 若最短数据帧长度增加 50 字节, 则允许最远两站增加 () (2021)
- (A) 80m (B) 40m (C) -80m (D) -40m
3. IEEE802.3 局域网中使用的 MAC 协议是 _____. (2022)
4. 一个采用 CSMA/CD 协议的局域网, 电缆最大长度为 1km, 无中继器, 数据传输速率为 1Gbps, 信号在电缆中的传播速度为 200000km/s, 则最短帧的长度为 64bytes. (2025)

两种 ALOHA 和三种 CSMA (2 道题)

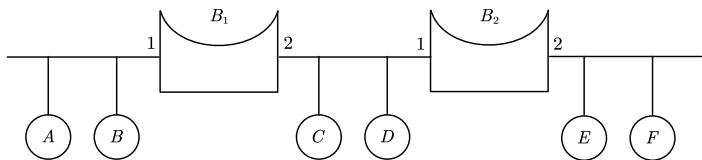
1. 以下哪一个属于使用了时槽周期的 MAC 协议? () (2022)
- (A) 非坚持型 CSMA (B) 1-坚持型 CSMA (C) p-坚持型 CSMA (D) 纯 ALOHA
2. P-坚持型 CSMA 必须在有时槽的情况下使用. (2023)



网桥, 交换机 (4 道题)

1. 网桥 (4 分) (2018)

如图, 三个局域网上用网桥 B1 和 B2 连接起来. 每一个网桥都有两个接口 (1 和 2). 系统开始运行之前, 两个网桥的转发表都是空的, 然后分别由以下各站向其他的站发送了数据帧: A 发送给 E, C 发送给 B, D 发送给 C, B 发送给 A, B 发送给 F. 请根据网桥自学习的原理将有关数据填写在下表中. (回忆版题目不完整, 新威考研找了一道类似的题)



发送的帧	B1 的转发表		B1 的处理 (转发?丢弃?登记?)	B2 的转发表		B2 的处理 (转发?丢弃?登记?)
	地址	接口		地址	接口	
A→E						
C→B						
D→C						
B→A						
B→F						

2. 在局域网中,解决多网桥互联的回路问题所采用的方式是 () (2020)
- (A)水平分裂算法 (B)生成树网桥 (C)新威考研 (D)公众号:新威考研
3. 在以太网中,下列那种方式可避免转发回路 () (2023)
- (A)水平分割 (B)生成树 (C)二进制指数退避 (D)汇集树
4. 以太网交换机收到一数据帧,经转发表对比该帧目的 MAC 地址不在表项中,则 () (2024)
- (A)将该帧转发给除接受该帧端口外所有端口 (B)将该帧传送给特定端口进行 ARP 查询
(C)将该帧发送给 HDPC 服务器 (D)丢弃该帧

无线局域网 (2 道题)

1. 无线局域网可采用 RTS/CTS 解决暴露站问题. () (2023)
2. 关于无线局域网,下列说法正确的是 () (2024)
- (A)为了减少冲突发生,无线局域网在发送信号时需要进行冲突检测
(B)无线局域网可采用 CSMA/CD 来进行冲突检测
(C)采用 RTS/CTS 方式,如果某一站点监听到 CTS 信号,但没有监听到对应的 RTS,那么该站点不需要为了避免冲突而静默.
(D)SIFS,DIFS,PIFS 等帧间间隔,主要是用于控制站点消息的优先级.



VLAN (1 道题)

1. 下列那种不是 VLAN 划分方式 () (2023)
- (A)基于端口 (B)基于 MAC (C)基于端口号 (D)网络地址

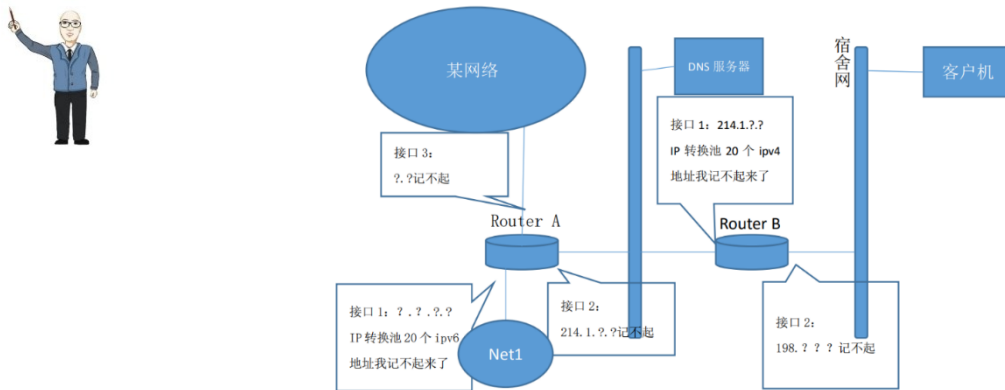
网络层 (共 22 道题)

网络层设备 (1 道题)

1. 哪一个网络设备工作在网络层 () (2022)
- (A)集线器 (B)路由器 (C)网桥 (D)中继器

DV,RIP (5 道题)

1. 一个网络路由 A 可以兼容 IPv4 和 IPv6, 题目也给出了 A 的路由表, 网络示意图如下: (2016)



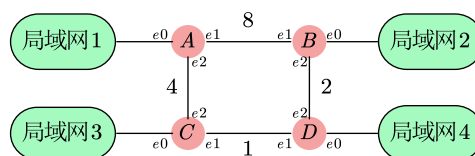
(1) A 会把哪个消息通过 RIP 协议发送给 B, 在发送的那条后面打√ (A 有三个接口, 每个接口都给出了 IP 地址, 但我想不起来了)

目的 IP 地址	发出端口	路径跳数	发送?
接口 1 发出的地址	1	5	
接口 3 发出的地址	3	1	
Router B 地址	2	1	

(2) 路由 B 和 A 交换了路由表, 问 B 路由表新增的条项是啥, 填入下表

目的 IP 地址	发出端口	路径跳数

1. 已知有如下网络, 边的权值表示花费. (2017)



(1) 若采用距离向量算法和水平分裂算法, 写出 D 节点收到的信息, 表格格式如下.

源节点	目的节点	花费

(2) 若采用距离向量算法, 写出 D 收敛后的转发表, 表格格式如下.

目的网络	下一跳	花费

3. 采用 RIP 协议, R6 的路由表如下: (2020) (原题回忆不起, 找一个类似的)

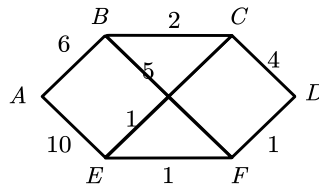
目的网络	距离	下一跳路由器
Net2	3	R4
Net3	4	R5

收到 R4 的更新信息如下:

目的网络	距离	下一跳路由器
Net1	3	R1
Net2	4	R2
Net3	1	直接交付

写出 R6 更新后的路由表.

4. 下图结点表示路由,边表示链路,数字表示距离. (2021)



(1) 所有路由采用距离向量路由算法,C 一启动收到来自

B:(<A,6>, <B,0>, <C,2>,<D,6>,<E,16>,<F,5>)

D:(<A,12>, <B,6>, <C,4>,<D,0>,<E,2>,<F,1>)

E:(<A,10>, <B,6>,<C,1>,<D,2>,<E,0>,<F,1>)



则 C 的路由表为? (格式如下:)

目的路由器	距离	下一跳路由器

5. 距离矢量算法被用于以下哪种网络路由协议 () (2022)

(A)OSPF

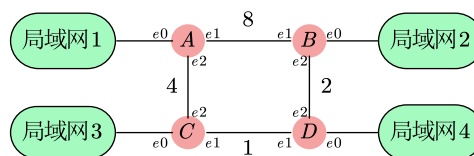
(B)IS-IS

(C)BGP

(D)RIP

LS,OSPF (2 道题)

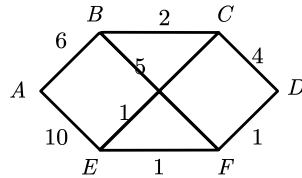
1. 已知有如下网络,边的权值表示花费. (2017)



若采用链路状态协议,写出 D 收到的链路状态信息,假设链路是无向的,即 A->B 和 B->A 等价,同一条链路只写一次即可,表格格式如下.

源	目的	花费

2. 下图结点表示路由,边表示链路,数字表示距离. (2021)



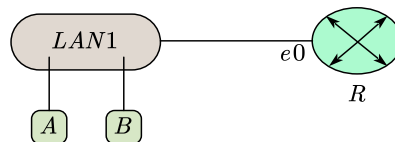
- (2) 若采用链路状态路由算法,给出 C 发出的链路状态信息
- (3) 如果网中两种算法 (DV 和 LS) 分别都用,路由稳定后,C 给出的路由表是否相同?

BGP (1 道题)

1. 为了增强互联网规模可扩大性,请分别从四个方面为 OSPF 和 BGP 提出优化建议. (2024)

路由转发 (2 道题)

1. 如图所示:A,B 两个主机在一个局域网里(LAN1:192.168.1.0/24). (2020)



网关都已配置为网关路由器 R 的地址 192.168.1.1,但是 A 的子网掩码错误配置成了 255.255.0.0,B 的子网掩码错误配置成了 255.255.255.240,没有其他配置错误.路由器表项如下:

网络地址	网络掩码	下一跳
192.168.1.0	255.255.255.0	直接交付
166.111.8.0	255.255.255.240	e1
166.111.8.128	255.255.255.224	e2
0.0.0.0	0.0.0.0	e3

现在有四个分组:

- (1) A 访问 166.111.8.100 (2) A 访问 166.111.8.200
- (3) A 访问 192.168.3.2 (4) B 访问 192.168.1.200 (假设存在)

一开始 A 和 B 的 ARP 缓存都为空,并认为 ARP 缓存时间足够大.问题如下:

- (1)写出上面 4 个分组发出时,都要用 ARP 查哪个 IP? 缓存是否命中? 是否会有回应? (在 ARP 协议部分做)

- (2)路由器能收到哪些分组? 匹配的路由表项是多少?

2. 局域网的路由表如下

地址/子网掩码	接口
192.168.160.0 /19	0
192.168.192.0 /18	1
192.168.128.0 /17	2
0.0.0.0	3

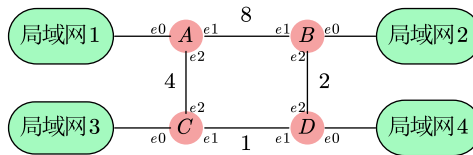


当目的地址分别是 192.168.64.18 和 192.168.172.17 的两个数据包到达此路由设备时,该设备将使用的转发接口分别是 () (2024)

- (A)1,0 (B)1,2 (C)3,0 (D)3,2

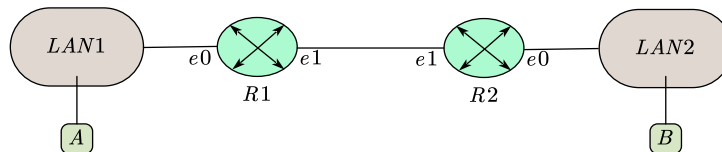
子网划分 (4 道题)

1. 已知有如下网络,边的权值表示花费。(2017)



若局域网 1 到局域网 4 分别有 78,38,14,4 台主机,请将网络 202.1.5.0/24 分配给图中局域网和路由器间的网段,写出划分后的网络,以及路由器端口 IP 地址及掩码。

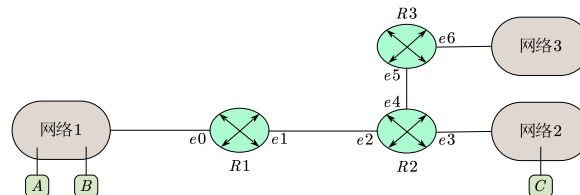
3. 如图所示,假设主机 A 与路由器 R1 相连,R1 与路由器 R2 相连,R2 与主机 B 相连。(2019)



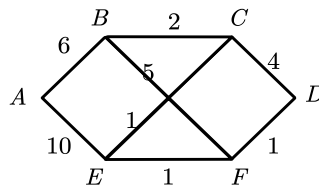
现有网络 161.111.1.0/24,需要将其 IP 地址分配给以下网络:LAN1、LAN2 以及路由器 R1 和 R2 之间的链路.要求使得 LAN1 和 LAN2 可用的 IP 地址数量之和最大,写出 LAN1、LAN2 的 IP 地址范围,R1、R2 的端口地址以及它们的地址掩码。

4. 如图所示,网络 1 有 100 台主机,网络 2 有 50 台,网络 3 有 20 台。(2019)

现有网络 166.111.4.0/24,将其划分给网络 1、网络 2,网络 3 及路由器 R1 和 R2 之间的链路、路由器 R2 和 R3 之间的链路,给出划分结果,并写出路由器接口 IP 地址.要求划分给网络 1、网络 2、网络 3 的 IP 地址数要不少于其包含的主机数。



5. 下图结点表示路由,边表示链路,数字表示距离 (2021)

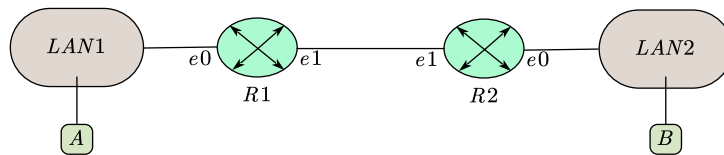


若 A-F 各连接 60、78、5、13、29、4 台机器的局域网,有网络 59.66.0.0/24,要求每台机器、连接局域网的路由器接口分得一个 IP 地址,设计一个划分方案,给出每个局域网的地址空间及其掩码。

IPv4 数据报分片 (1 道题)

1. 如图所示,假设主机 A 与路由器 R1 相连,R1 与路由器 R2 相连,R2 与主机 B 相连.假定链路 A-R1 可支持的最大帧长为 1024 字节,其中包括 14 字节的帧头;链路 R1-R2 可支持的最大帧长为 512 字节,其中包括 8 字节的帧头;链路 R2-B 可支持的最大帧长为 512 字节,其中包括 12 字节的帧头。

(2019)



若 A 要发送一个数据段大小为 900B, TCP 报头为 20B 的报文, 在网络层加了一个 20B 长的 IP 分组头部, Identification 的值为 x, 问这个 IP 分组在 A~R1, R1~R2, R2~B 上传输时, 分组的 Total length、Identification、DF、MF、Fragment Offset 的值各是多少?

广播地址 (2 道题)

1. 166.111.67.8/21 所在网络的广播地址为 () (2021)

(A) 166.111.79.255 (B) 166.111.71.255 (C) 166.111.67.255 (D) 166.111.64.255

2. 166.111.67.8/21 的广播地址是 _____. (2022)

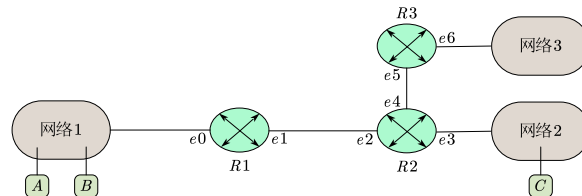


ARP 协议 (2 道题)

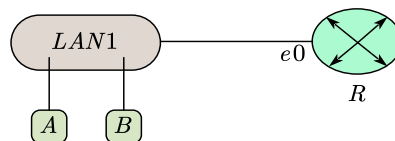
1. 如图所示, 网络 1 有 100 台主机, 网络 2 有 50 台, 网络 3 有 20 台. (2019)

(1) 简述 AB 通信时和 AC 通信时使用 ARP 协议的具体情况.

(2) 当 A 发送报文给 C 时, 需要经过 A→R1, R1→R2, R2→C 三个阶段, 写出各段上报文的源 IP, 目的 IP, 源 MAC, 目的 MAC (用 MAC-A, IP-A, MAC-e0 等表示) .



2. 如图所示: A, B 两个主机在一个局域网里 (LAN1: 192.168.1.0/24). (2020)



网关都已配置为网关路由器 R 的地址 192.168.1.1, 但是 A 的子网掩码错误配置成了 255.255.0.0, B 的子网掩码错误配置成了 255.255.255.240, 没有其他配置错误. 路由器表项如下:

网络地址	网络掩码	下一跳
192.168.1.0	255.255.255.0	直接交付
166.111.8.0	255.255.255.240	e1
166.111.8.128	255.255.255.224	e2
0.0.0.0	0.0.0.0	e3

现在有四个分组:

(1) A 访问 166.111.8.100 (2) A 访问 166.111.8.200

(3) A 访问 192.168.3.2 (4) B 访问 192.168.1.200 (假设存在)

一开始 A 和 B 的 ARP 缓存都为空,并认为 ARP 缓存时间足够大.问题如下:

(1)写出上面 4 个分组发出时,都要用 ARP 查哪个 IP? 缓存是否命中? 是否会有回应?

3. 发送 ARP 请求消息时,采用 () 方式 (2024)

(A)单播 (B)多播 (C)组播 (D)广播

虚电路 (1 道题)

1. 电路交换和虚电路分组交换共同点是 () (2022)

I.分组有序;II.设备端口带宽预约占用;III.不会产生拥塞

(A)I,II,III (B)I,II (C)I,III (D)I

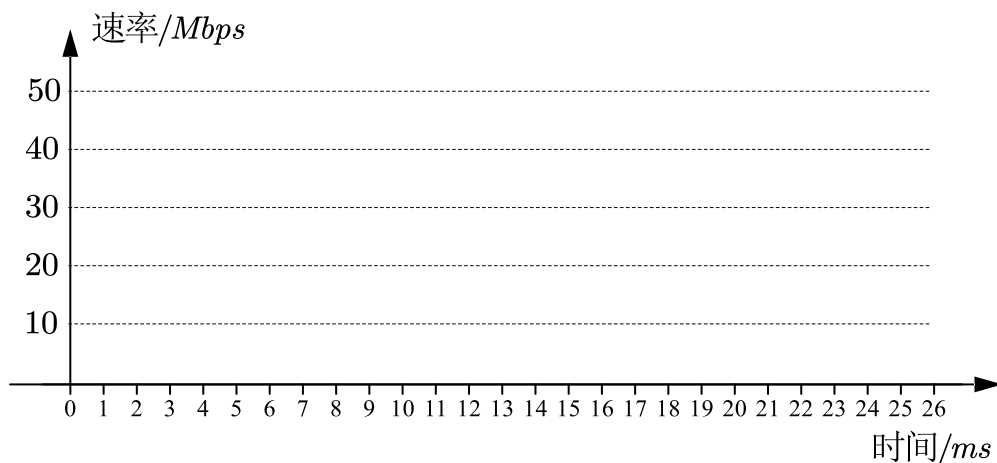
令牌桶 (1 道题)

1. 解答题 (6+4=10 分) (2025)

某运营商宽带套餐提供最大 50Mbps 的上行带宽,上行方向采用令牌桶算法对用户发出的数据进行流量整形,令牌桶的大小为 40kb,令牌产生的速率为 10Mbps,初始时令牌桶是满的.假设信道传播延迟和协议开销等忽略不计,网络不发生拥塞或误码.某用户从第 0ms 开始每隔 10ms 都会产生 100kb 的数据,并立即发送.题中, $k=1000$, $M=1000000$.

1) 请参考下图画出前 25ms 内用户发送数据的速率随时间变化的折线图,标出各折线点的坐标,并写出计算过程;

2) 当时间趋于无穷大时,该用户的平均数据传输速率是多少?



传输层 (共 9 道题)

TSAP (1 道题)

1. TCP 的 TSAP 的标识是 () (2021)

(A)IP+端口号 (B)MAC+端口号 (C)IP+MAC (D)MAC+IP+端口号

TCP 协议基本性质 (1 道题)

1. TCP 是有连接的,可靠的,端到端的.面向数据报的协议. () (2025)



TCP 连接管理 (1 道题)

1. TCP 连接管理中通过_____与定时器结合来实现释放连接. (2022)

TCP 可靠传输 (1 道题)

1. 主机 A 和主机 B 之间采用 TCP 传输,A 向 B 传输 3 个 TCP 段,其有效载荷长度分别为 200,300,500 字节,第三个 TCP 段起始序号为 800,A 向 B 传输第一个段成功,第二个段丢失,则主机 B 在收到第三个段后,返回的确认序号是 () (2024)
- (A)200 (B)500 (C)800 (D)1000



TCP 拥塞控制 (5 道题)

1. TCP 传输过程中,拥塞窗口大小为 W (W 比较大) 时发生拥塞,每次都以最大发送段长 MSS 发送数据,发送方有足够多的数据要发,求出平均速率近似是多少 () (2019)
- (A) $0.75W * MSS / RTT$ (B) $0.5W * MSS / RTT$ (C) $0.875W * MSS / RTT$ (D) $0.625W * MSS / RTT$

2. TCP 拥塞控制 (2016)

A 向 B 发送 TCP 报文数据,初始序号为 0,初始阈值为 16KB,每次都以 $MSS=1KB$ 的长度发送报文.

- 1) A 一直向 B 发送数据,未遇超时情况,当 A 收到确认号为 8192 的报文后,拥塞窗口的大小为多少?
- 2) A 一直向 B 发送数据,未遇超时情况,当 A 收到确认号为 31744 的报文后,拥塞窗口的大小为多少?
- 3) 当遇到超时 (窗口大小是 40KB 的时候超时) 后,新阈值和此时窗口大小为多少?

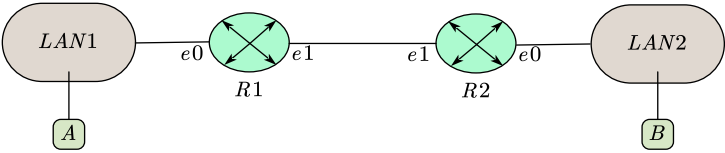
3. TCP 使用 reno 算法进行拥塞控制,初始阈值为 400KB,接收窗口大小为 600KB, $MSS=100KB$,填写下列表格: (2020)

事件	拥塞窗口大小	阈值	第 i 轮所发送的包序号 (单位:MSS)
---	100KB	400KB	(1)0
收到(1)中所有包的确认			(2)
收到(2)中所有包的确认			(3)
收到(3)中所有包的确认			(4)
收到(4)中所有包的确认			(5)
(5)中发送的第 1 个包超时			(6)

4. 相比于 TCP 的 reno 算法,请列举 CUBIC 和 BBR 算法的改进性. (2024)

5. 现在 A 向 B 传输 7 个 TCP 报文,报文长度等于 MSS ,采用 reno 进行拥塞控制,初始阈值是 $16 * MSS$,假设接收窗口足够大,且网络无拥塞和差错,且 B 每收到一个报文就回复对应的确认报文,那么从开始建立连接到 A 收到最后一个确认报文结束共经历了多少 RTT? (2018 大题的第 3 小问) (原本

回忆版模糊成啥了,新威考研强势修复)



应用层 (14 道题)

WWW (1 道题)

1. 解释以下 URL 各部分的意义 <http://info.tsinghua.edu.cn:80/index.jsp> (2019)



字段	意义
http	
info.tsinghua.edu.cn	
80	
/index.jsp:	

HTTP (3 道题)

1. 某用户通过 HTTP1.1 非流水线方式请求访问一个网页,网页一共有个短文本和 5 个 jpg 图片,问该用户从请求连接到能看到整个网页为止共经过 () 个 RTT? (2016)

(A)8 (B)7 (C)6 (D)5

2. 计算题 (10 分) (2023)

假定客户主机与 web 服务器间的瓶颈链路容量为 10 分组/秒,TCP 链接管理分组和 HTTP 请求分组的发送时间和协议栈处理时间均忽略不计,web 服务器支持的并发链接数与并发请求数均大于 5,TCP 拥塞控制算法任何时间都可以充分利用带宽客户浏览器下载封装在 10 个分组的 HTML 页面文件,页面内嵌多个服务器本地资源文件,可以封装在 20,30,40 个分组中传播.

服务器最大速率 10 个分组每秒,往返时延 1 秒,TCP 并发请求均大于 5,每个网页大小 10 个分组,其中包含 3 个服务器本地资源文件,资源文件可以分别通过第 20,30,40 个分组发送

- 1) 采用 http1.0 时,用户下载本地资源和文件 (所需要的时间为多少) 最短响应时间多少
- 2) 采用 http1.1 时,用户完成下载本地资源和文件 (所需要时间为多少) 最短响应时间多少
- 3) 采用 http2.0,平均需要多少响应时间完成页面和资源文件下载最短需要多少时间 (响应时间定义为各资源文件下载完成时间的平均值),那么这 3 个资源文件下载完成的最小响应时间是多少?

3. HTTP,QUIC,SMTP 协议在传输层采用的协议分别为 () (2024)

(A)TCP,UDP,TCP (B)TCP,TCP,TCP (C)UDP,TCP,UDP (D)UDP,UDP,UDP

电子邮件 (2 道题)

1. 关于 SMTP 协议,下列说法中正确的是 () (2016)

(A)服务器不能给客户发送邮件 (B)客户不能接收服务器的数据



(C)一个 TCP 通道不能发送多封邮件

(D)服务器之间不能互相传送邮件

2. 客户端代理发送邮件所采用的协议 () (2023)

(A)SMTP

(B)POP3

(C)IMAP

(D)SNMP

域名和 DNS (7 道题)



1. 以下正确的选项是 () (2018,2020)

(A)不同地方的人在双十一访问淘宝购物,对应的服务器 IP 地址可能不同

(B)DNS 资源记录是通过二元组的形式

(C)DNS 通过集中式数据库提供服务

(D)访问 WWW.TSINGHUA.EDU.CN 和 www.tsinghua.edu.cn 不一样

2. 如域名 info.tsinghua.edu.cn 对应的 IP 地址为 166.111.4.98,解释为何会发生如下现象: (2019)

(1) 访问 http://info.tsinghua.edu.cn/index.jsp 正常,而访问 http://166.111.4.98/index.jsp 异常.

(2) 访问 http://166.111.4.98/index.jsp 正常,而访问 http://info.tsinghua.edu.cn/index.jsp 异常.

3. 关于 DNS,下列说法中正确的是 () (2021)

I.一个域名可以对应多个 IP 地址

II.一个 IP 地址可以对应多个域名

(A)I

(B)II

(C)I 和 II 都不对

(D)I 和 II 都对

4. 域名解析时递归查询不一定会访问根域名服务器. (2023)

5. DNS 缓存会造成 () (2023)

(A)可以提高响应性,没有一致性问题

(B)可以提高响应性,有一致性问题

(C)不能提高响应性,没有一致性问题

(D)不能提高响应性,有一致性问题

6. 域名服务提供了域名到 IP 和端口号相互转换的机制,例如 www.tsinghua.edu.cn 可解析为 166.111.4.100 和 80. () (2025)

SNMP (1 道题)

1. 以下关于 SNMP 协议的说法中,错误的是 () (2017)

(A)SNMP 协议具有性能管理,故障管理,配置管理,记账管理和安全管理五个功能

(B)SNMP 采用 TCP 协议进行管理

(C)公众号: 新威考研

(D)www.xinweikaoyan.com

