**数据结构**

**大O 渐进时间复杂度**

**选择判断：**

**1.(logn)^n= θ(n^logn)。** (2020年)

* 正确
* 错误

**2.n^(logloglogn)=O(⌊logn⌋!)。** (2019年)

* 正确
* 错误

**3.T(n)=a>0，无论常数a多大，时间复杂度为T(n)=T(n/2)+O(1)的解总是O(logn)。** (2018年)

* 正确
* 错误

**4.若f(n)=时间复杂度O(g(n)),也不一定有f(n)=O(g(n-1))。** (2017年)

* 正确
* 错误

**栈、向量**

**选择判断：**

**逆波兰表达式0!1+23!4+^\*56!7\*8!?/-9+值为2017。**(2020年)

A.?处为加号

B.?处为减号

C.?处为乘号

D.?处为除号

**后缀表达式扣去一个符号来猜扣去的是什么，跟2017的类似。** (2018年)

**一个非法表达式，问强行求解的值是多少。** (2018年)

**有如下逆波兰式结果为2016，问?中的运算符号是多少（ ）** (2017年)

**2  0 ！ \*  2  2  \*  6  +  ^  18  8  ?  9  /  \***

A.+   B.\*

C.^   D. !  E./

**一组输入 MAMAMIA 入栈，要求出栈顺序也为 MAMAMIA，共有几种方案？** (2016年)

A 4 B 5 C 6 D 7

**简答**

**逆波兰表达式的优点是什么？既然中缀转换为逆波兰消耗的时间就可以计算出表达式值，那逆波兰意义何在？**(2019年)

**函数的调⽤栈中如果有相同的函数，则他们必然紧邻。**(2019年)

* 正确
* 错误

**Fib查找时以前后黄金分割点作为轴点的常系数相同。** (2018年)

* 正确
* 错误

**排序**

**基于CBA的算法对所有大小为n的数组时间复杂度是Ω(nlogn)。** (2018年)

* 正确
* 错误

**就地算法的空间复杂度是（ ）** (2018年)

 A.O(1)      B.      C.      D.

**冒泡排序**

**在进行冒泡排序时，有可能出现某些元素在排序过程中一直远离它的最终位置。** (2016年)

* 正确
* 错误

**插入排序**

**插⼊排序每次插⼊数据，即使不增加循环节，也不⾄减少。**(2019年)

* 正确
* 错误

**交换某个逆序对中的两个元素，必然会减少总逆序对数。**(2019年)

* 正确
* 错误

**简答：**相比选择排序，插⼊排序的优势是什么，说明两点。(2019年)

**拓展：**

选择排序和冒泡排序相比的优势是什么

快速排序相比归并排序的优势是什么

**序列（64，63，...，2，1）进行直接插入排序比较次数最接近于（ ）** (2017年)

A.2800   B.2600  C.2400   D.2200   E.2000

**折半插入算法在寻找插入的位置时，采用的是二分查找，因此整个折半插入算法的时间复杂度为O(nlogn). 。** (2016年)

* 正确
* 错误

**基数排序**

**若底层排序算法不稳定，采用基数排序算法后（ ）**(2020年)

A.未必正确并且不稳定

B.正确并且不稳定

C.未必正确并且稳定

D.正确并且稳定

**如果基数排序底层采⽤不稳定的算法，那么得到的结果可能是不正确的。**(2019年)

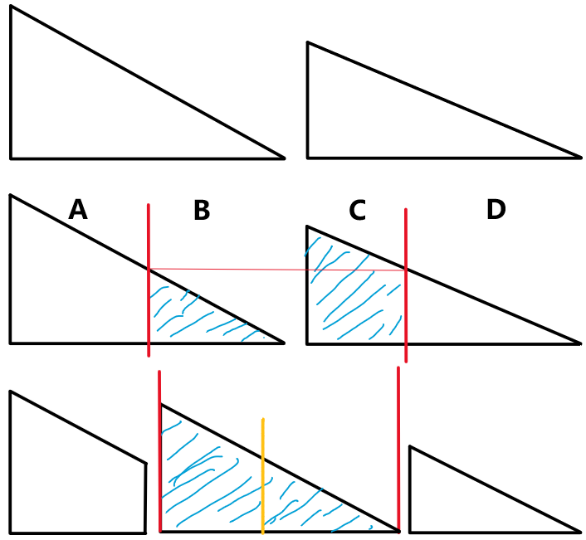
* 正确
* 错误

**基数排序的底层排序算法一定是稳定的（ ）** (2018年)

* 正确
* 错误

**归并排序**

**有序向量的二路归并排序** (2017年)



改进的归并策略为：

* 将两个子序列分别进行分割为四个子序列，使得两个序列在割点处的值相等。（似乎也可能是确定割点为1/2或1/3处）
* 直接拼接中间的两个序列（即交换它们的位置），从而得到3个有序子序列。
* 对这3个子序列，先归并1、2，用得到的结果再归并3，从而完成整个策略。

1）填空 merge

2）对 ABCDE 处的注释补充

3）rotate（）

4）说明这种算法的优缺点

**快速排序**

**快速排序的时间复杂度在平均情况下为O(nlogn)，最好情况亦如此。**(2020年)

* 正确
* 错误

**基于比较式的算法可以在O(n)时间内在任意n个无序整数中找出前10%。**(2020年)

* 正确
* 错误

**二分查找**

**任何情况下折半查找都比顺序查找快。** (2018年)

* 正确
* 错误

**算法题**

**单峰向量(13')** (2018年)

已知A[0,n ), A[0~k)严格单调递增，A[k~n)严格单调递减，设计一个O(logn)算法找出k

1)伪代码描述算法

2)说明算法正确性

3)证明最坏情况下时间复杂度也是O(logn)

**树**

**卡特兰数**

**\_个无差别节点构成的真二叉树，与由2019对括号构成的合法表达式一样多。**(2020年)

A.2018 B.2019 C.4038 D.4039

**叶节点为2019的真⼆叉树数量小于2018对括号所组成的合法表达式数量。**(2019年)

* 正确
* 错误

**五个互异节点构造的二叉搜索树有多少种？（ ）** (2017年)

A.  B.  C.  D.

**哈夫曼编码**

**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年)

A.最大长度为6

B.最大长度为7

C.最大长度为8

D.最大长度为9

**交换哈夫曼树不同深度的节点，平均编码长度必然改变。**(2019年)

* 正确
* 错误

**PFC(最优前缀编码)互换不同深度节点位置一定会破坏其性质。** (2018年)

* 正确
* 错误

**哈夫曼树距离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年)

* 正确
* 错误

**AVL树**

**规模为n的AVL一次插入操作最坏情况下会引起logn次局部重构。** (2020年)

* 正确
* 错误

**在 AVL 树中，（）可能会发生两次旋转调整？** (2016年)

A 添加、删除节点操作

B 仅删除节点操作

C 仅添加节点操作

D 添加、删除节点都不

**对于同一个长度为 n 的序列分别按照递增和递减的顺序构造 AVL 树，那么“存在正整数k，使 n=2^k-1 ”是“两次构造的堆相同”的（ ）**(2018年)

A.充分不必要条件     B. 必要不充分条件     C. 充分必要条件     D. 既不充分也不必要条件

**将关键字1，2，3...，2016插入初始为空的平衡二叉树中，假设只有一个根节点的二叉树高度为0，那么最终二叉树的高度是多少？（ ）**(2017年)

A9.       B.10      C.11       D.12

**Splay伸展树**

**对于不符合局部性原理的访问，splay的分摊复杂度不是O(logn)。**(2019年)

* 正确
* 错误

**伸展树插入操作的分摊时间复杂度O(logn)** 。(2018年)

* 正确
* 错误

**B树**

**7阶B-树根节点常驻内存，则对规模为2017的B-树最多需要几次访外存？（ ）**(2018年)

A.      B.          C.          D.

**搜索7阶B树的第2016个关键字，假设B树根节点在内存中，则共需启动几次I/O。（ ）** (2017年)

A.      B.          C.          D.

**红黑树**

**红黑树上所有节点的黑深度和黑高度之和必相等。**(2020年)

* 正确
* 错误

**试举出红黑树优于AVL树的场景，红黑树相比AVL树的优势是什么？**(2019年)

**拓展：**

伸展树相比AVL树的优势是什么

**BST二叉搜索树**

**二叉搜索树中最大的节点（ ）** (2016年)

A 仅有左孩子，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

**叶节点数量为2018的⼆叉树，层次遍历时队列容量必然小于2018。**(2019年)

* 正确
* 错误

**对于二叉树，通过先序遍历和后序遍历不能确定其层次遍历。**(2019年)

* 正确
* 错误

**判断与证明 给定一棵二叉树的先序和后序遍历序列，通过先序和后序遍历序列能否确定唯一层次遍历序列？若可以给出证明，不可以则说明理由。**(2020年)

**判断与证明 给出中序序列{D B A E C F}和层次序列{A B C D E F}能否唯一确定一颗二叉树？能给出步骤，不能的话请构造其中一棵**(2016年)

**算法题 ：下图给出二叉树的结构体声明**：A,B,C,D和F都是3分每题，E为5分。 (2020年)

struct BinNode { //二叉树节点

BinNodePosi(T) lc;

BinNodePosi(T) rc;

int size; //当前节点和孩⼦总数

}

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，模拟中序遍历不得分）；

D.splyto(a, x) ，a为x的祖先，通过zigzag操作将x调整变为a的孩子，若a为NULL，x调为根节点

E.将一颗splay树调成完全二叉树（可用上述函数），要求时间复杂度为O(nlogn)，迭代深度不超过O(logn);

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。

**算法题： 返回后序遍历的第 K 个节点，时间复杂度不超过 x 的深度，Ο(depth(x))** (7+3+4 = 14) (2019年)

struct BinNode{

int size; //当前节点和孩⼦总数

BinNode \*lchild,\*rchild;

};

BinNode \*rank(BinNode\* t,int k){

//有效代码⾏数不超过 12 ⾏

//不要尝试模拟后序遍历，时间复杂度会超时。

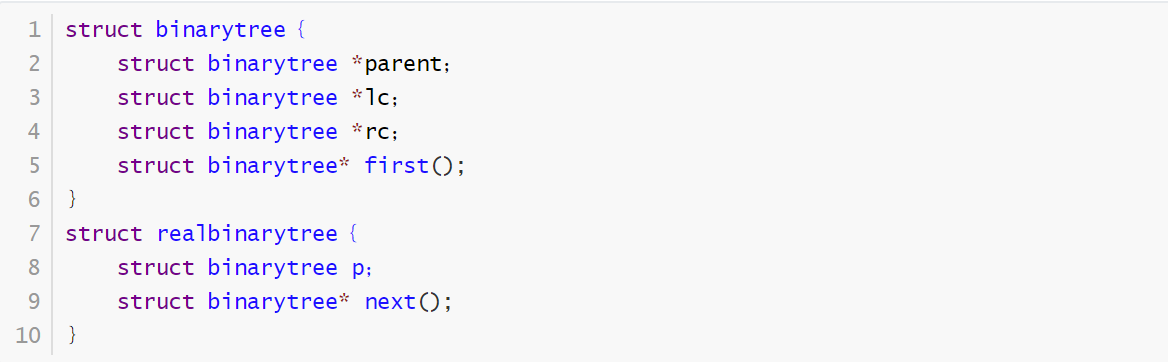
}

⼀，给出具体算法实现。

⼆，解释你的算法。

三，分析时间复杂度和空间复杂度。

**算法题：若二叉树的数据结构如下** (2017年)

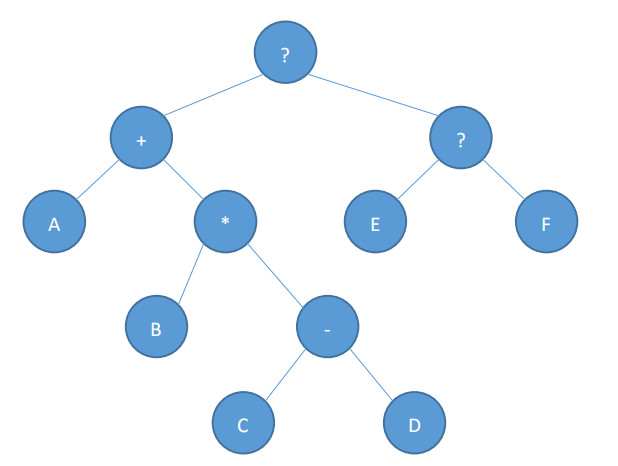


（1）若first()函数是取二叉树后序遍历节点的第一个节点，请写出first（）函数代码。

（2）若next（）函数是取该节点的后序遍历的后继，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，证明遍历时间复杂度为o(n)。

**算法题：设计一个算法，把一个中序遍历 ABCD-\*+EF??(后面三个符号忘记了不过不重要)构造成如下图所示的二叉树** (2016年)



a) 描述算法思想

b) 伪代码实现

**算法题：给了一个算法，问访问节点的顺序，树的样子是一颗深度为 4 的二叉树。** (2016年)

typedef struct binNode {

char data;

struct binNode \*rc,\*lc;//左子树右子树

}binNode;

twist(node x) {

if(!x) return;

if(x->rc) {

twist(x->lc->rc);

putchar(x->data);

twist(忘了);

putchar(忘了);

}

else{

Twist(忘了);

Putchar(忘了);

}

}

**堆**

**完全二叉堆**

**完全二叉堆删除操作平均时间复杂度为O(1),最坏情况下为O(logn)。**(2020年)

* 正确
* 错误

**将n个元素组成一个完全二叉堆，时间复杂度至少为O(nlogn)。**(2020年)

* 正确
* 错误

**如果插⼊的关键码独⽴均匀分布，堆的插⼊操作平均时间复杂度为O(1)。**(2019年)

* 正确
* 错误

**输入随机的情况下完全二叉堆的插入平均时间是常数。** (2018年)

* 正确
* 错误

**胜者树-败者树**

**败者树删除操作的时间复杂度在常系数上优于胜者树。**(2020年)

* 正确
* 错误

**简答 相比于锦标赛树，败者树的优势是什么？**(2019年)

**左式堆**

**Crane算法合并左式堆A和B为H，H右侧链节点未必都来自A或B右侧链。**(2020年)

* 正确
* 错误

**左式堆最右侧链长度为k，则左式堆\_\_\_\_\_\_含有\_\_\_\_\_\_个内部结点。（ ）** (2018年)

A.最少 2^k  B.最少 2^k-1  C.最多 \*\*  D.最多 \*\*

**词典-哈希、散列**

**选择判断：**

**开放式散列比封闭式散列可以更有效地利用局部缓存。**(2020年)

* 正确
* 错误

**简答：相比开散列，闭散列的优势是什么，说明两点。**(2019年)

**对长度为m=4k+3素数的散列表双平方探测一定能访问其全部元素 。** (2018年)

* 正确
* 错误

**散列表用不超过长度的素数，即使分布理想，使用取余法仍然会堆积。** (2017年)

* 正确
* 错误

**散列长为2017，采用单平方探测，已经存入1000个元素，问此时最多有（ ）个懒惰删除的桶单元。** (2018年)

A.8  B.9  C.  D.

**简答：散列表长为 13，采用双散列函数解决冲突：** (2016年)

散列表长为 13，采用双散列函数解决冲突：H(key) = key % 13，H’(key) = ( 7 \* key % 10 ) + 1。

输入顺序为：12，23，45，57，20，03，78，31，15，36。

【注：回忆版题面已按习题解析对应题面修正，实际考试中可能有所简化】

1) 构造散列表

2) 求等概率下搜索成功的平均查找长度

**跳转表**

**在n个节点的跳转表中，单个词条的期望塔高是θ(logn)。**(2020年)

* 正确
* 错误

**图**

**选择判断：**

**有向图DFS后有k条边被标记为后向边，图中未必恰含k个环路。**(2020年) 重要，易错

* 正确
* 错误

**有向无环图DFS后各节点\_\_按拓扑排序。**(2020年)

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

**简答：DFS过程中何时标记前向边？何时标记后向边？**(2019年)

**权值都为正整数的图能用迪杰斯特拉构造出最短路径。**（2016年)

* 正确
* 错误

**简答：对于稠密图，迪杰斯特拉算法使用多叉堆替换⼆叉堆，为什么？多叉堆分叉数m怎么确定？**(2019年)

**算法题**

**请利用图的广度优先遍历找出图中的最小环，若不存在环则输出+oo,要求时间复杂度为O(n\*e)，空间复杂度为O(n)，最小环即环中边数最少的环。** (2017年)

（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。

**简答：程序应该是 prim 算法**，问是否能够构成最小生成树，如果能就证明，不能举出例子驳斥。

V 表示图的点集，U 表示已经确定路径的点集，初始时 U 为空，F 为已经确定的路径，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，具体算法就请翻书吧(2016年)

**字符串**

**KMP**

**单选：随机英文字母串匹配，最好情况下蛮力\_\_\_\_\_\_\_ KMP，平均复杂度蛮力 \_\_\_\_\_\_\_ KMP。（ ）**(2020年)

A.坏于 坏于

B.相等 坏于

C.坏于 相等

D.相等 相等

**判断：在字符集各字符出现概率相同时，kmp算法时间渐进程度接近蛮力算法。** (2017年)

* 正确
* 错误

**简答：在何种情形下KMP优于蛮力算法，为什么？**(2019年)

**单选：模式串HHFBHHFHHFBSHF改进后的next表，以下正确的是（ ）**(2020年)

A.next[13]=1, next[0]=-1

B.next[13]=1, next[0]=0

C.next[13]=0, next[0]=-1

D.next[13]=-1, next[0]=-1

**单选：即使不使⽤改进的next表，kmp依然可以达到线性的时间复杂度。**(2019年)

* 正确
* 错误

**没改进的next算法时间复杂度也是O(n)。** (2018年)

* 正确
* 错误

**BM算法**

**对于长度为 m 的串进行串匹配时好后缀数组中 gs[0]=1 的概率为（ ）** (2018年)

A.1/m   B.1/2^(m-1)

 C.1/2^m   D.1/2^(m+1)

**算法题**

**最大和区间(13')** (2018年)

给定一个整数序列，求出连续子序列和的最大值

1)说明算法思路

2)伪代码描述算法

3)说明时间复杂度和空间复杂度

题注(大致意思)：蛮力算法就不要用啦，是O(n^3),只有设计出O(n)算法才有可能满分，O(n^2)酌情给分。

**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** (2016年)

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)

a) 描述算法思想

b) 伪代码实现

c) 计算程序的算法复杂度。

**操作系统**

**物理内存，虚拟存储**

**Belady现象**

**判断**：FIFO算法存在belady现象。 (2020年)

**证明**：(6分) LRU、BEST、CLOCK、FIFO 页面置换算法是否能产生 belady 异常，若可以举出例子，不可以给出证明。 (2018年)

**多选题**：以下会发生 belady 异常的是（） (2017年)

a）FIFO 算法

b）LRU 算法

c）CLOCK 算法

d）LFU 算法

e) 改进 CLOCK 算法

**判断**：x86开启了⼆级页表，则⼀级页表可以不在内存中。 (2019年)

**判断**：ucore时钟中断为10ms，故不能完成 10ms以下的定时。 (2019年)

**判断**：中断向量表中存放着中断门的优先级。 (2019年)

**判断**：Buddy 算法中，释放一个空间后可以根据起始长度和大小与相邻空闲空间合并。 (2018年)

**填空**：CPL<=DPL[门]和CPL>=DPL[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年)

**填空**：x86-32CPU的硬件组成，CR3寄存器⽤于存储页⽬录表起始 。 (2019年)

**选择**

以下由硬件完成的是( ) (2017年)

A.获取中断源

B.形成中断入口地址

C.Eax 寄存器保存

D.

以下算法会产生很多不必要的小碎片的分区（ ） (2017年)

a）最佳匹配算法

b）首次适应算法

c）最坏匹配算法

能够有效避免产生小碎片的算法是（） (2017年)

a）最佳匹配算法

b）首次适应算法

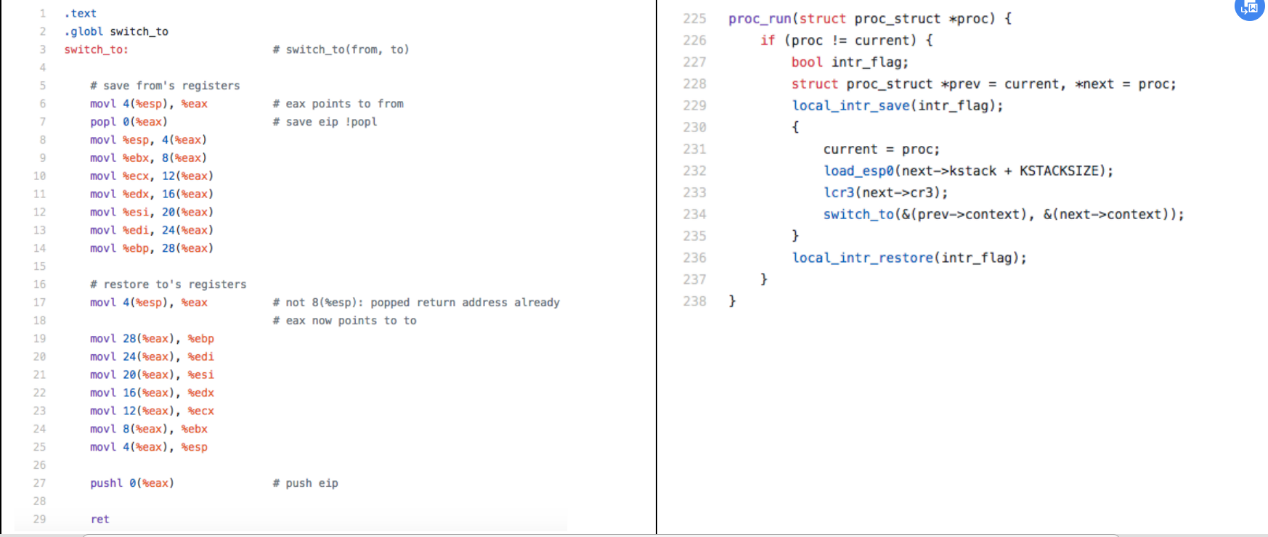
c）最坏匹配算法

忘了就记得选项（ ） (2016年)

C 有些数据 cache 并不能存放

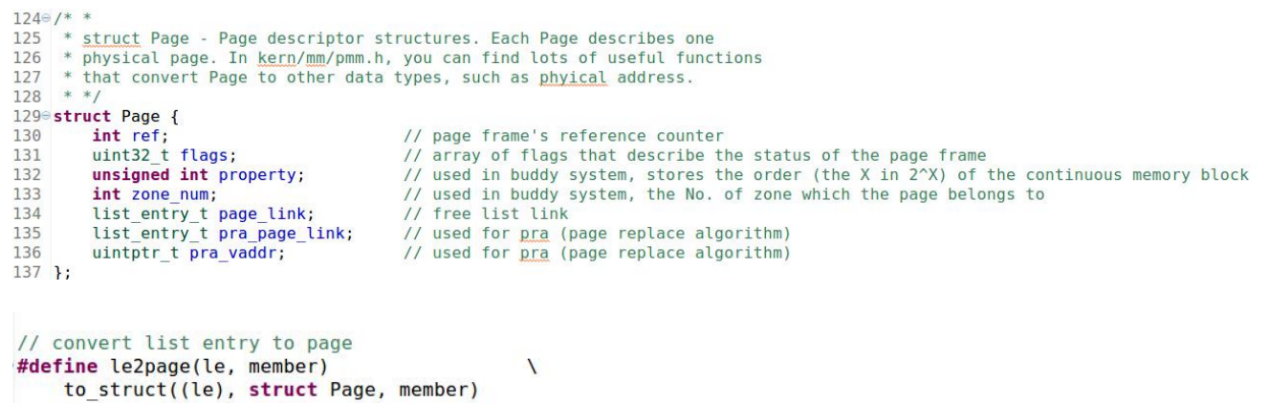
D 多核处理器共享一个高速缓存

**简答题：**（6分）UCORE在以下代码中找到页表切换，堆栈切换，switch\_to函数获得两个函数参数的代码部分并注释 (2019年)



**简答题：**（4分）内存分析。给了段内存的 dump，以及⼀些页表项的信息，分析 A、B 进程的逻辑地址转换 (2019年)

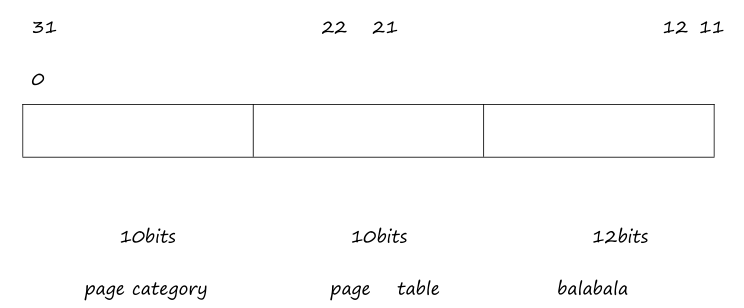
**简答题：**（6分）ucore (2018年)





le2page(\*page,page\_link)语句都需要展开那些宏定义？说明这个语句的含义。(还有一段 ucore 代码是 buddy 算法的页面分配函数，好像跟这道题关系不大，就不贴了主要是没找到。)

**简答题：**Intel X86-32 CPU 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 X86-32 CPU 上运行。(2018年)



**简答题：**一道关于 ucore 的题目 (2017年)

1）很多代码 balabala，求 intr

2）flag，turn balabala 填一行代码

**简答题：**一台计算机虚拟空间 8KB，物理空间 4KB，二级页表，页表项 2B,页目录项 1B，页表大小32B，求进程页面大小有多少 b (2017年)

**进程和线程**

**判断**

只有⼀个main函数的程序不能有多个线程。 (2019年)

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年)

操作系统采用 copy on write 机制时，fork()函数会复制进程的页目录表。 (2018年)

8 位 cpu 不能进行线程切换。 (2016年)

**填空**

⽗进程先退出，还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年)

父进程退出后，没结束的子进程变成 。 (2018年)

**选择**

exec（）系统调用会改变以下哪些参数（ ） (2017年)

A.进程 ID

B.父进程 ID

C.文件打开指针

D.？

关于线程和管程错误的是（ ）(2017年)

多重选择题（ ） (2016年)

程序和进程的区别

A 程序啥啥啥

B 进程啥啥啥

C 进程是动态啥啥啥

D 程序是动态啥啥啥

**处理机调度**

**判断**

死锁的充分必要条件是互斥，持有并等待。 (2020年)

SJF调度算法可能出现饥饿现象。 (2020年)

信号量机制可以解决程序死循环的问题。 (2020年)

不安全状态就是死锁状态。 (2019年)

死锁必要条件：互斥访问，占有并等待，非剥夺，循环等待。 (2016年)

管程就是一个黑箱子，程序员往里面扔函数，同一时间只有一个函数在执行。 (2018年)

管程和信号量在功能上等价。 (2018年)

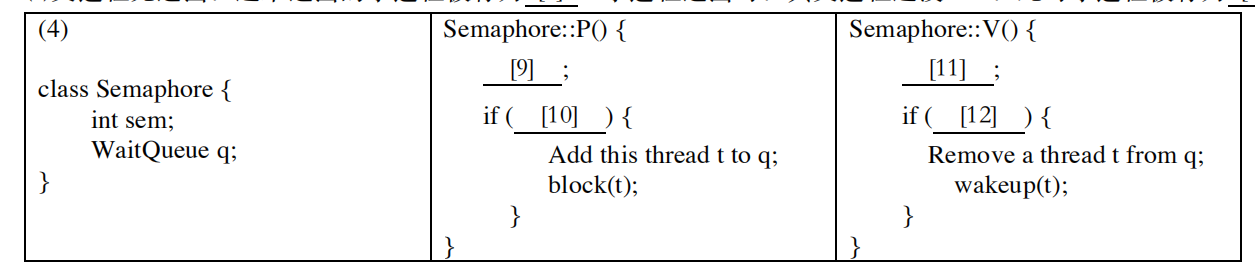
管程将资源抽象成条件变量，通过变量值的增减来控制进程的访问。 (2018年)

使用自旋锁不能保证进程按先来后到的顺序使用 cpu 资源。 (2018年)

**填空**

stride调度算法中，进程分配时间和其优先级成反⽐，优先级越低进程上CPU运⾏的时间就会更长，⽤⼋位⽆符号数来表⽰进程A的运⾏时间有可能会溢出，但是当步长s 127时，只要做⼩⼩的技巧就仍可以正确判断时间，然后又引⼊进程B，a为A的stride，b为B的stride，当sign(a-b) 时说明A正在运⾏，此时a+s 256，当sign(a+s-b) 0时，轮到B运⾏。(2019年)

(2019年)



高响应比调度算法的分子是 ，分母是 。 (2018年)

优先级反置指的是 抢占了 的资源， 时低优先级进程能动态改变优先级 (2018年)

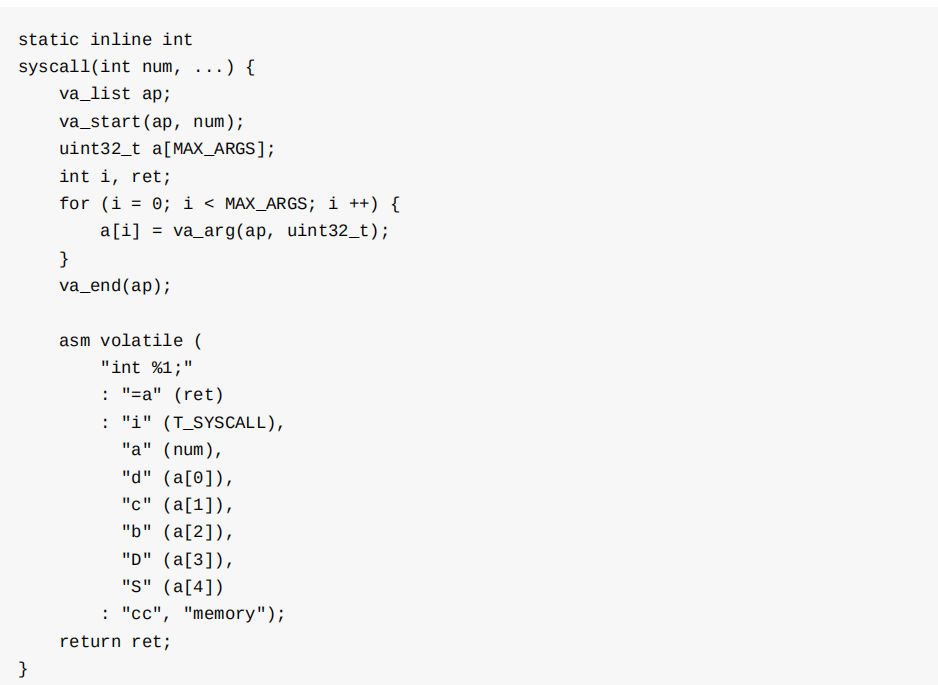
支持暂时放弃互斥资源访问权，等待信号 (2018年)

**系统调用**

**简答题**

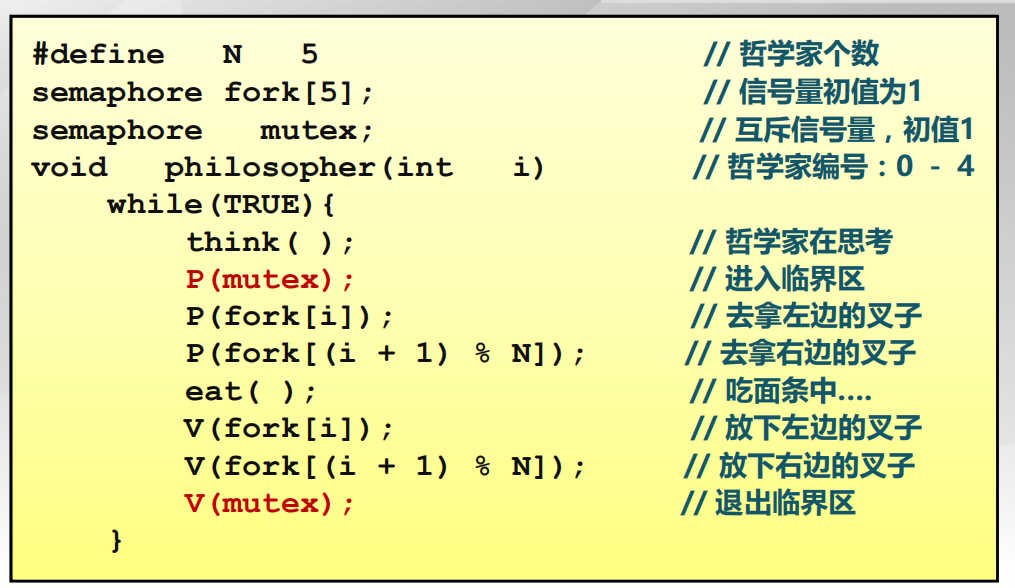
1. 给出系统调用的四个分类，例如xx类完成xx的创建、撤销和退出 （2020）

2. 给出getpid系统调用的执行过程



以上系统调用允许的最大参数个数为 ， 其中num是 ，edx和ecx存储的分别是第几个参数，系统调用的返回值存放在\_\_\_\_\_。

**哲学家就餐问题** (2018年)



1)上述算法会不会死锁，如果会请举例

2)算法是否允许两个哲学家同时进餐，若可以请举例

**文件系统**

**判断：**删除一个文件，该文件的所在的当前目录将改变。 (2020年)

**判断：**PCB中的当前工作目录可以加速文件的查找。 (2020年)

**选择**

设文件F1的当前引用计数值为1，先建立文件F1的符号链接（软链接）文件F2，在建立文件F1的硬链接F3，然后删除文件F1。此时，文件F2和文件F3的引用计数分别是（ ）。 (2020年)

A是计数为1的⽂件，创建A的硬链接B，再创建A的软连接C，在创建B的硬链D，B计数 , D计数 。 (2019年)

UNIX索引结构存放的位置是（ ） (2020年)

A. 超级块     B. 索引节点     C.     D. 忘了

以下哪种磁盘阵列存取速度快（ ） (2017年)

A.RAID0

B.RAID1

C.RAID4

D.RAID5

**简答题**

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（糟糕，忘记是索引表还是指针表了）一个存储块为 8KB，一个指针 4B，问理论上这个文件系统能存放的最大文件是多大？用 TB+GB+MB+KB+B 表示 。(2016年)。

**I/O子系统**

**判断**

延迟写操作可以减少对磁盘的访问次数。 (2020年)

最短寻道时间算法在SSD存储设备中无效。 (2020年)

**计算机组成原理**

**运算器**

**判断(1\*5')**

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年)

整数Int x<0,则必有-x>0 。(2020年)

假设x类型是C语言中的int，若x>0，则x\*x>0。 (2019年)

C 语言若 int x,y 若 x>y，则-x<-y 。 (2018年)

冯诺依曼结构体系中把程序也当做数据放在内存中。 (2019年)

**填空题(2+2+3+3=10)**

十进制整数+1234的32位的补码是：\_\_\_\_\_\_\_。（16进制，小端机表示） (2019年)

-2017 的 32 位补码表示\_\_(16 进制或 2 进制)。 (2018年)

十进制单精度浮点数-27.625在IEEE754浮点标准下表示：\_\_\_\_\_\_\_。（16进制） (2019年)

-2017 的 IEEE 单精度浮点表示\_\_\_\_\_\_\_。 (2018年)

若海明码 **P1P2** D1 **P3** D2 D3 **P4** 为**01** 0 **1** 1 0 **0**，则该海明码有\_\_\_\_\_\_\_位错误（0 位，1位，2位），正确的 D1D2D3 为\_\_\_\_\_\_\_。(2017年)

**选择 ：**IEEE规格化单精度浮点数能表示的最小正数是（ ） (2017年)

**控制器**

**判断(1\*5')**

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年)

CPI越短，程序执行能力越快。 (2020年)

CPI减少，执行相同程序的时间也减少。 (2019年)

提高 cpu 主频可以加快程序执行速度。 (2018年)

MIPS五级流水线设计中，使用充分设置功能单元的方法可以改善结构冲突 (2019年)

**填空题**

指令由操作码和\_\_\_\_\_\_\_组成。 (2017年)

MIPS五级流水线中，解决数据冲突的方法，给出3个. (2019年)

处理机\_\_\_\_\_\_\_逻辑电路进行算术运算，\_\_\_\_\_\_\_逻辑电路可以用于数据暂存，\_\_\_\_\_\_\_逻辑电路用于分支选择。 (2018年)

**选择**

下列表述中正确的是（ ） (2020年)

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.我忘了

解决流水线的数据冲突不可以采用的是（ ）(2020年)

A. B.

C.分支预测 D.静态调度

以下哪种不可以解决数据冲突（ ）(2018年)

A.暂停流水线

B.分支预测

C.调整指令顺序

D.数据旁路

MIPS五级流水中，有哪个数据冲突（ ） (2019年)

A.RAR B.RAW C.WAR D.WAW

以下关于五段流水线的处理机说法错误的是（ ） (2018年)

A.多个处理器不会发生结构冲突

B.每个周期执行一个功能

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

以下哪个不是响应异常的处理（ ） (2018年)

A.保存 pc

B.保存通用寄存器

C.保存异常原因

D.恢复 pc

计算机运行的最小单位（） (2017年)

 a）？ b）微指令

c）指令 d）？

**解答** (2020年)

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事件，其余未算入)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IF | ID | EX | MEM | WB |
| 250ps | 180ps | 150ps | 300ps | 200ps |

实现以下3条指令：

addu rd,rs,rt

lw rt,rs,imm

j target

1. 按照单周期、多周期、流水线设计，最短的时钟周期为？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？

**计算题(5)** (2019年)

MIPS 处理器pc输入延迟2ns，寄存器堆输出延迟2ns，内存延迟 10ns，ALU 延迟 6ns，寄存器堆输入延迟1ns，流⽔线寄存器以及多周期锁存器输出延迟为 2ns，要有计算过程。

addu rs rt rd

subu rs rt rd

ori rs rt imm

lw rs rt imm

sw rs rt imm

beq rs rt imm

j target

1.按照单周期设计，指令内存与数据内存分开，计算指令延迟？

2.按照多周期设计，指令内存和数据内存在同⼀个内存模块，最长和最短的指令延迟分别是指哪条指令，分别计算对应的延迟，以及所需时钟周期数。

3.按照五级流⽔线设计，指令内存和数据内存不在同⼀个内存模块，处理器能达到的最大主频是多少？

**解答题** 五段流水线，每段 10ns，每个寄存器 5ns，以下一段程序(4 句)，问执行时间是多少 (2018年)

lw \*\*\*

sub \*\*\*

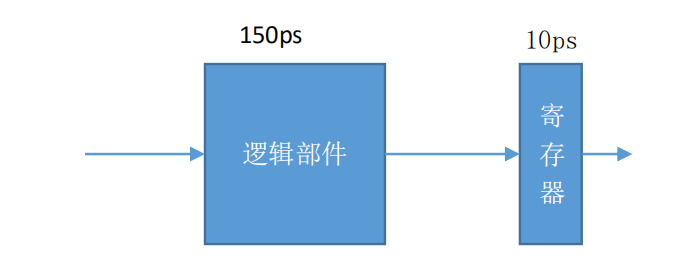
and \*\*\*

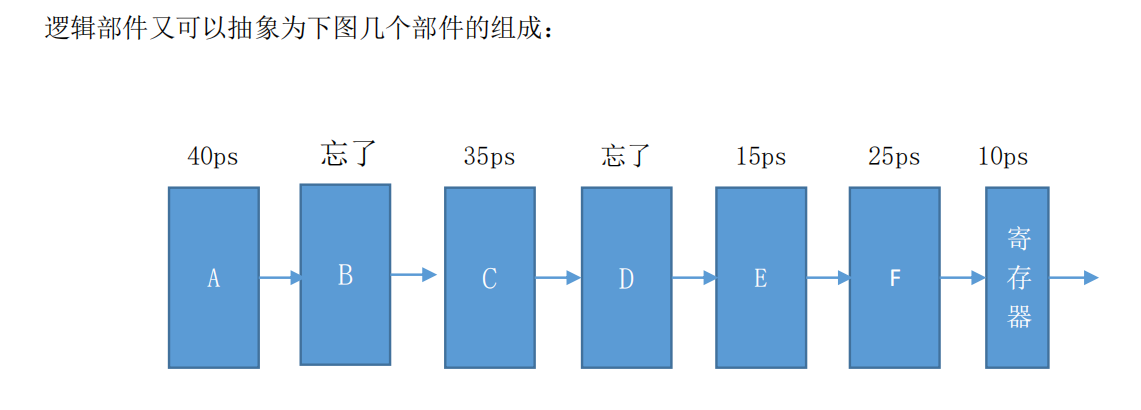
or

**解答题：**指令流水线可能发生的冲突分类，以及原因。 (2017年)

**流水线大题**(2016年)

任何指令的执行部件可以抽象为一个逻辑组件和寄存器的结构，逻辑组件的延迟为 150ps，寄存器为 10ps，示意图如下

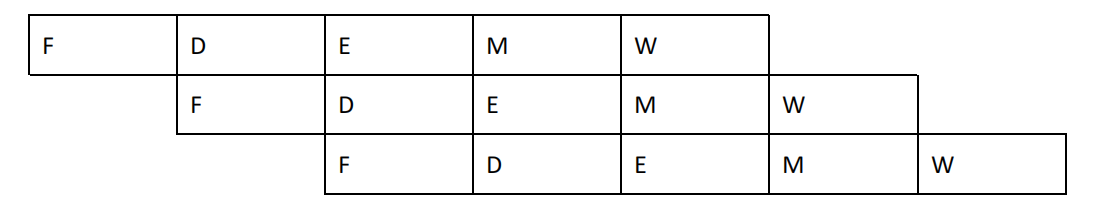




1） 要形成一个 4 级流水，应该将三个寄存器安插在那些位置？问该四级流水的延迟和最大吞吐率。

2） 为达到最大的吞吐率应该设计成几级流水？寄存器应该安插在哪些位置？问该流水的延迟和最大吞吐率。

如果将上面的部件形成五级流水，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



3） 以下三个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？（每条指令所需要的上一条结果的数据都要等到上一条运算的结果才能进行）

4） 以下四个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？四条指令一共用了多少个时钟周期？（每条指令所需要的上一条结果的数据都要等到上一条运算的结果才能进行）（程序都想不起来了囧，就记得前两条是把数字移入寄存器，第三条的执行需要前两条数据，最后一条需要第三条的数据）

**存储器**

**判断：**缓存原理利用了程序的局部性 (2020年)

缓存缺失的类型包括\_\_\_\_\_\_\_、\_\_\_\_\_\_\_、\_\_\_\_\_\_\_。（写3个） (2019年)

**选择**：全相联，4路组相联，2路组相联中缓存命中率最高的是 (2020年)

高速缓存器的几种映射方式\_\_\_\_\_\_\_、\_\_\_\_\_\_\_、\_\_\_\_\_\_\_。 (2018年)

**填空** 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，计算直接映射和二路组相连的缓存命中率 (2020年)

以下说法正确的是（ ） (2018年)

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. D.

**解答题** (2017年)

30 位虚拟地址，28 位物理地址，一级页表，页大小 16KB，访问 5ns；Cache 采用直接映射，大小64KB，块大小 4B，访问 5ns；主存访问 40ns。

1）虚拟页表脏(dirt)位 1 位，有效位 1 位，问页表大小

2）cache 标记位，索引位，块内地址各多少位

3）一次 cache 命中访问时间，cache 失效访问时间，命中率为 90%平均访问时间

4）系统进程切换时以下操作是否需要，原因

a）清除 cache 有效位

b）将已经调入页表清空

5）注意到页表访问和 cache 访问时间相同，可否通过修改 cache 映射方式，使 cache 和页表一同访问，可以的话做出相应设计，并计算 cache90%命中率的时候的平均访问时间。

**判断：**对于传统机械硬盘，读100MB数据，顺序读取时间小于随机读取时间。 (2019年)

**判断：**RAID6 坏两个磁盘也可以工作。 (2018年)

下列哪一项没有容错能力 (2019年)

A. RAID0 B. RAID1 C. RAID5 D. RAID6

**填空：**100MB的数据，RAID1为,RAID5为 (2020年)

**填空**

一台计算机显示器的分辨率为800\*60，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年)

DMA使用总线的方式为\_\_\_\_\_\_\_和\_\_\_\_\_\_\_。 (2017年)

**选择：**MIPS中断中不是由硬件负责的是（ ） (2020年)

A. 开中断

B.保存通用寄存器

C.保存异常原因

D.关中断

下列关于静态存储器和动态存储器的描述正确的是（ ） (2019年)

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

下列哪个是对的 （ ） (2019年)

A．虚拟内存空间比实际的地址空间大

B．虚拟内存空间比实际的地址空间小

C．虚拟内存空间连续存放，实际内存一定连续存放

D．虚拟内存空间不连续存放，实际内存有可能连续存放

**输入输出**

下面总线说法哪个正确（ ） (2019年)

A.并行总线速度大于串行

B.异步总线速度大于同步

C.单总线速度大于双总线

D.以上说法均错误

**计算机网络**

**物理层**

**选择题**

以下设备只工作在物理层的是 （ ）(2020)

选项是路由器、中继器之类的设备

数据啊在模拟电路中传播需要的设备是（ ） (2020)

A. 调制解调器

B. 编码解码器

电话⽹络和TCP⽹络的性质的⽐较，电话⽹应该是建⽴电路 (2019)

奈奎斯特定理适用于以下哪些场合 （ ） (2018年)

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

**数据链路层**

**选择题**

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)

蜂窝移动⽹络六边形，频率840HZ，则每个点可⽤最⼤频率（ ） (2019)

以太网中最短帧长1000bit，两点相距100m,数据在光纤中的传播速率为2\*10^8m/s,问最大发送速率 (2019)

A.1Gb/s B.2Gb/s C.100Mb/s D.200Mb/s

停等协议通信线路利用率最低的是（ ）(2019)

A. 源和目的之间距离很近，速度快

B. 源和目的之间距离很近，速度慢

C. 源和目的之间距离很远，速度快

D. 源和目的之间距离很近，速度慢

两地相距3000公里(传播速度6us/公里) 最大帧64字节，采用GBN协议，带宽为1.536Mbps，则若要最大限度发挥网络带宽，至少需要多少比特的序号（ ）(2018年)

A.4 B.5 C.6 D.7

选择重传协议，序号为 0-13，发送窗口为 7，当数据发送不产生冲突是，接收窗口最大值为多少（ ）(2018年)

A.4 B.5 C.7 D.8

**解答题**

太空站的传输速率为128kbps，发送 512 字节帧，端到端的传输延迟 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，吞吐量位多少？若要使信道利用率达到最大，则帧序号至少为多少？ (2017年)

**网络层**

**选择题**

计算⼀个计算分组时间，两个转发加两个传播 (2019)

**解答题**

**解答题** (这一题比较长，可能会有若干数字记错了) (2020)

A,B 两个主机在一个局域网里（192.168.1.0/24），网关都已配置为网关路由器的地址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。

**现在有如下4个 ip 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(这个数字记不清了，但是用 200 也是可以做题的)

**路由器表项如下：**

1, 192.168.1.0 255.255.255.0

2, 不记得了，应该没用

3, 不记得了，应该没用

4, 166.111.8.0 255.255.255.240

5, 166.111.8.128 255.255.255.224

6, 0.0.0.0

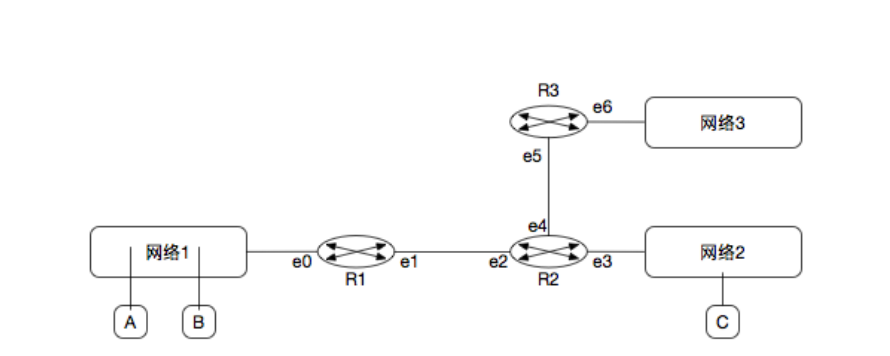
一开始 A和B的ARP缓存都为空，并认为ARP缓存时间足够大。问题如下：

(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

(2), 路由器能收到哪些分组？匹配的路由表项是多少？

(3)距离向量的更新，D收到了来自B和C的信息，D到B和C的距离分别为2和3(具体数字可能反了)，D中的路由表项为

**⼦⽹划分** (2019)



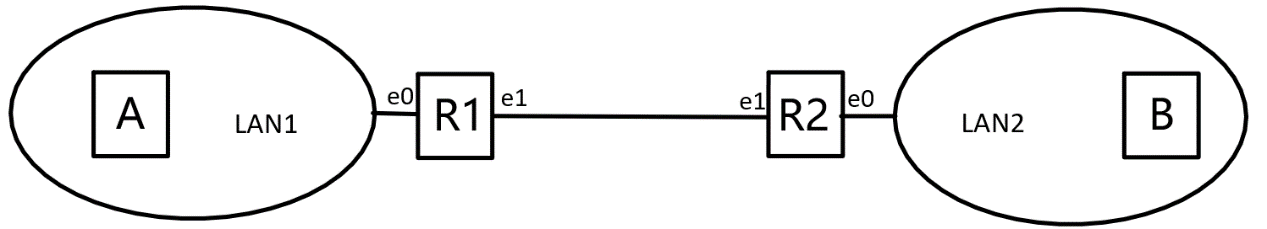
(1) 如图，⽹络1有100台主机，⽹络2有50台，⽹络3有20台，请将166.111.4.0/24划分给⽹络并写出路由器接口ip

(2) 简述AB通信时与AC通信时使⽤ARP协议的具体情况

(3) 当A发送报⽂给C时写出各个段上报⽂的源IP，⽬的IP，源MAC，⽬的MAC（⽤MAC-A,IP-A,MAC-e0等表⽰）

**路由器+TCP大题（10’）** (2018年)

一道路由器大题，两个路由器，三段网络的最大（数据部分）帧长度分别为1024,512,912，报头长度分别为14,12,12。两个路由器 R1、R2，R1 的 e0 端口链接着局域网 LAN1，R2 的 e0 端口连接着局域网 LAN2，R1 的 e1 端口连接着 R2 的 e1 端口。主机 A 在 LAN1 内，主机 B在 LAN2 内。如图：



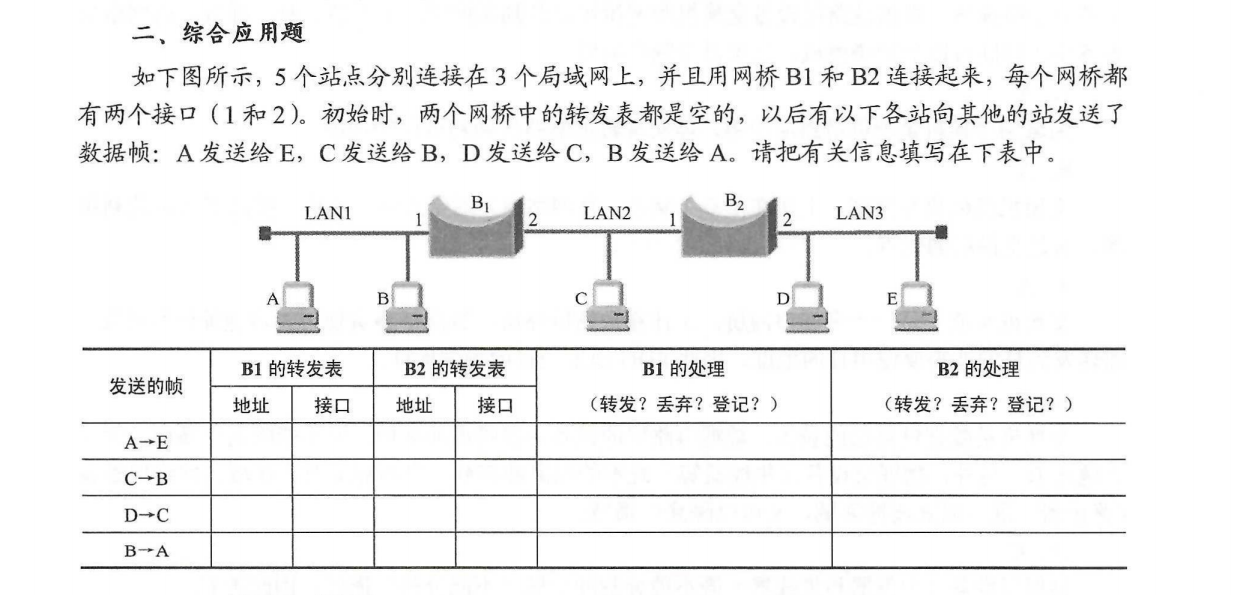
1）现在有 IP 地址 161.111.1.0/24 分给这些网，包括 R1 和 R2 的两个端口，问该如何分配，才能使 LAN1 和 LAN2 所获得的 IP 地址数量之和最多，写出 LAN1、LAN2 的 IP 地址范围，R1、R2 的端口地址以及它们的子网掩码。（4’）

2）若 A 要发送一个数据段 900B，TCP 头部 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，求分组的 Total length、Identification、DF、MF、Fragment Offset 的值各是多少？（4’）

3）若从 A 到 B 所需往返传输时间为 RTT，现在 A 要向 B 传输 7 个 TCP 报文，那么从开始建立连接到 A 收到最后一个确认帧结束共经历了多少 RTT？

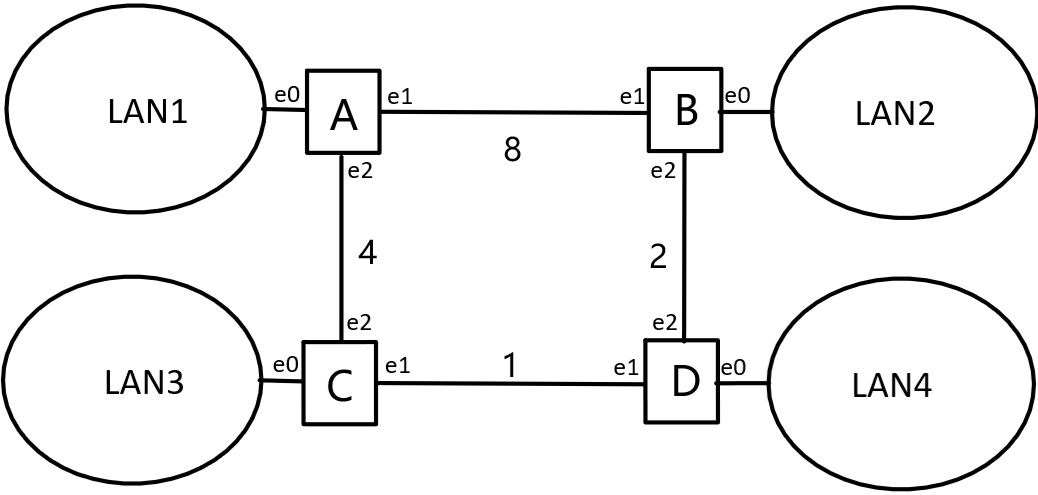
**网桥大题**：两个网桥三段子网，建立转发表，要求填表。（和**王道**上的一道原题几乎一样。） (2018年)

下面是《王道计算机网络-》3.8.4 的综合题。



已知有如下网络，边的权值表示花费。  (2017年)

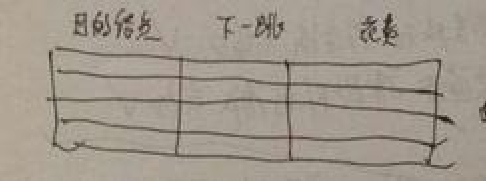
给一个网络的表填写距离向量表和路径表，ip 地址为 200.1.5.0/24 四个局域网分别有 78，38，14，9 个主机，划分子网，每个路由器的端口，网络地址范围。



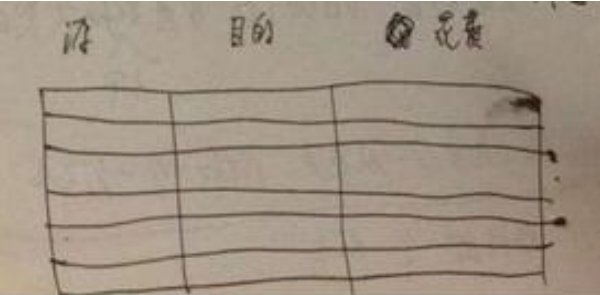
（1）若采用距离向量算法和水平分裂算法，写出 D 节点收到的信息，表格如下。



（2）使用 RIP 算法写出 D 收敛后的转发表，表格格式如下。

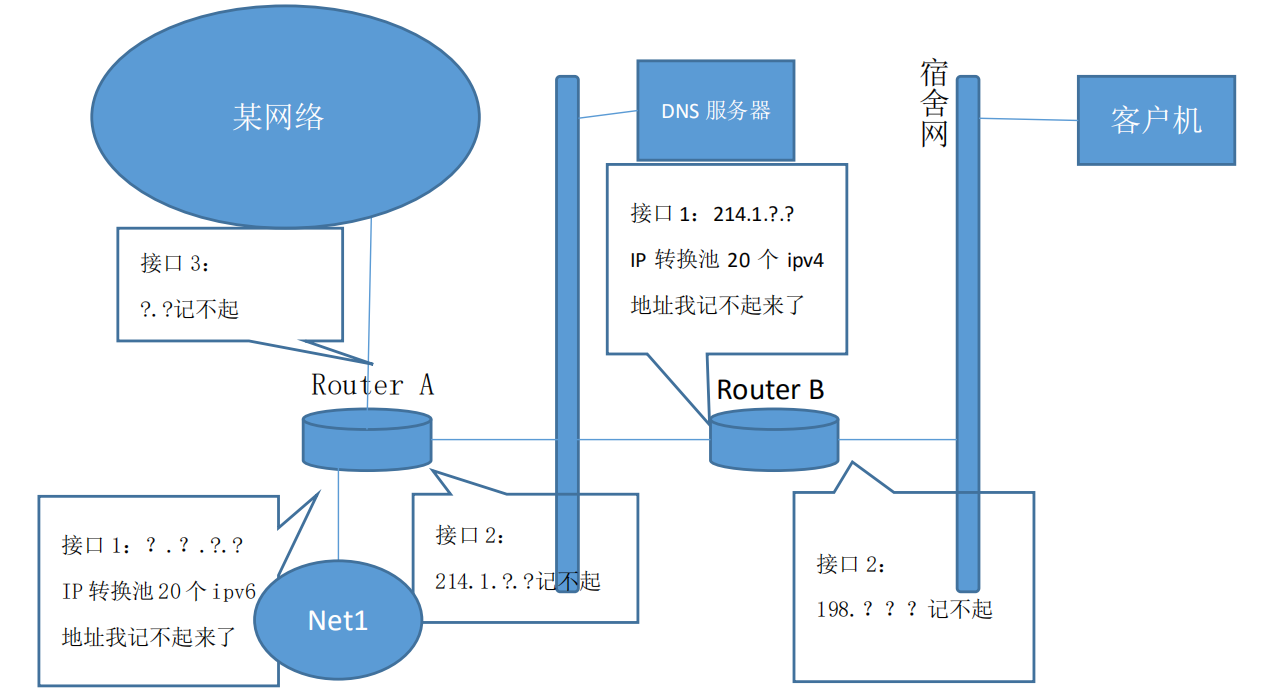


（3）若采用链路状态协议，写出 D 收到的链路状态，若表格中源和和目的等价，即 AB 和 BA 等价，表格格式如下。

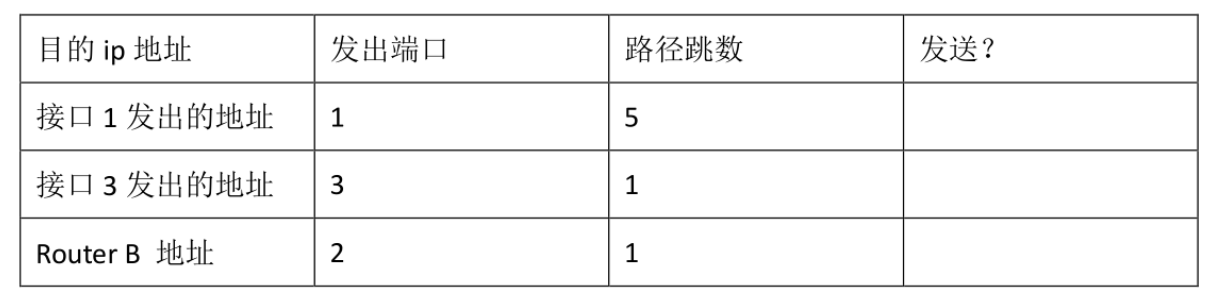


（4）若局域网 1 到 4 分别有 78，38，14，4 台主机，请将网路 202.1.5.0/24 分配给图中局域网和路由器间网段，写出划分后的网络，以及路由器端口 IP 地址及掩码。

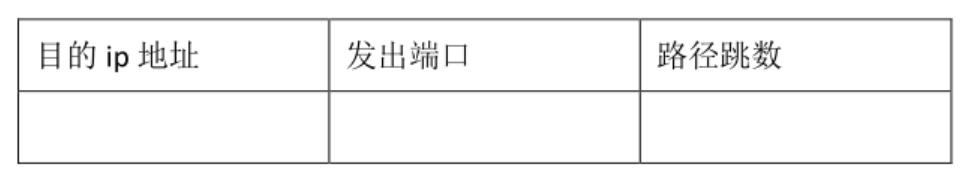
**解答题：**一个网络路由 A 可以兼容 IPv4 和 IPv6，题目也给出了 A 的路由表，网络示意图如下 (2016年)



1）路由 B 发 RIP 协议给 A，问 A 会把那个消息发送给 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）



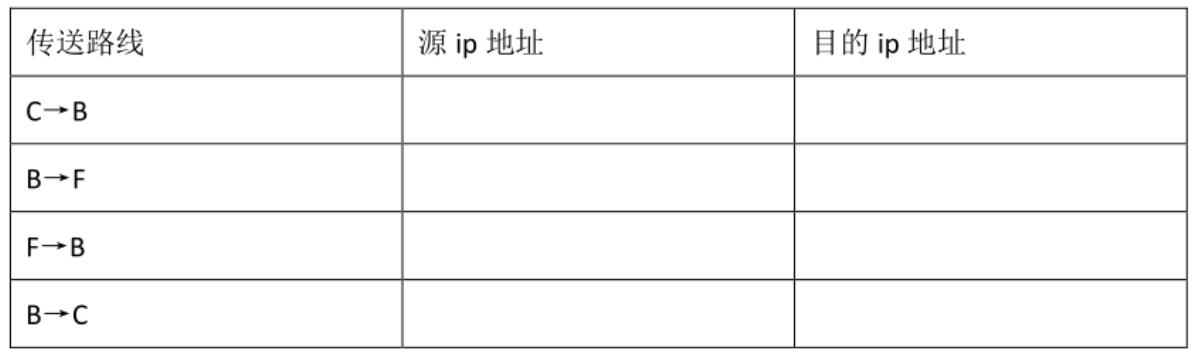
2）路由 B 和 A 交换了路由表，问 B 路由表新增的条项是啥，填入下表



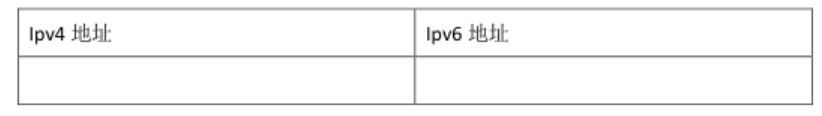
路由 B 具有 NAT 功能，宿舍管理员因此给宿舍电脑分配了一批 ip 地址，（都是 10.9 打头的，具体想不起来），如果客户机要访问 dns 服务器，B 就会从 ip 转换池里挑出一个 ip 地址与该客户机的 IP 地址进行映射，比如将 ip 池中的 224.1.1.7（想不起来我自己编的）分配给客户

机，那么其他网络就通过 224.1.1.7 访问该客户机，而不是客户机自己的 ip 地址。A 的转换方式也是如此。

3）填写下表中 ip 数据报的源地址和目的地址，C 代表客户机，F 代表 DNS 服务器，B 代表路由 B



4） 路由 B 要经过 A 来访问 IPv6 网络，问 A 中 B 的映射表项内容，只需给出一个可能项即可



**传输层**

**选择题**

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. C.生成树网桥 D.

TCP中，拥塞窗口大小W（窗口为m时发⽣拥塞），最大发送段长MSS，给RTT，发送⽅有⾜够多的数据要发，求算出平均速率近似是多少（ ） (2019)

TCP/IP 与 OSI （） (2018年)

A.

B.OSI 从上到下依次是应用层，会话层，表示层，网际层，网络层，数据链路层，物理层

C.TCP/IP 从上到下依次是应用层，网络层，数据链路层，物理层

D.TCP/IP 适用场合比 OSI 更广

**TCP 拥塞的典型题目** (2020)

TCP使用慢启动算法，初始阈值为400KB，接收方接收窗口大小为600KB



**TCP拥塞控制大题** （2016）

关于拥塞避免的，A向B发送数据，阈值为16KB，MSS=1KB

1. A一直向B发送数据，未遇到超时情况，当A收到了ACK为8KB的报文后，拥塞窗口的大小为多少？
2. A一直向B发送数据，未遇到超时情况，当A收到了ACK为31KB的报文后，拥塞窗口的大小为多少？
3. 当遇到超时后，（好像是又经过了多少RTT记不太清了）新阈值和此时窗口大小为多少

**应用层**

**选择题**

DNS，和18年专业课表述基本相同（ ） (2020)

A.双十一淘宝购物，不同地方的人得到的IP地址可能不同

B.DNS存储IP是通过二元组的形式

C

D.数据库集中存储

**dns相关问题** (2018)

A.天猫双12购物，不同地方两个人访问淘宝得到的ip一定相同

B.

C.存储ip是五元组

D.数据库集中存储

以下关于 SNMP 协议说法错误的是（ ） (2017年)

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

B.SNMP 采用 TCP 协议进行管理

C. D.

SMTP 协议正确的是（ ） (2016年)

A 服务器不能给客户发送邮件（记不清了）

B 客户不能接收服务器数据

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

用户用http请求访问一个网页，网页一共有一个短文本和5个jpg图，问用户从请求连接到能看到整个网页为止共经过（ ）个RTT？ (2016年)

A 8 B 7 C 6 D 5

**解答题**

2. 简答题（2+1+1=4）

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

(2) 如域名info.tsinghua. edu.cn对应的ip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