逻辑思维题

1. 过桥：小明一家五口人要过桥，一次只允许两个人通过，且如果两个人过桥的话，只能按着走路慢的人的速度来走。如果他们一个一个过桥的话，小明需要15秒，妹妹要20秒，爸爸要8秒，妈妈要10秒，奶奶要23秒。请问他们怎样过桥时间最短？

2. 谁做对了？

甲、乙、丙三个人在一起做作业，有一道数学题比较难，当他们三个人都把自己的解法说出来以后，甲说：“我做错了。”乙说：“甲做对了。”丙说：“我做错了。”在一旁的丁看到他们的答案并听了她们的意见后说：“你们三个人中有一个人做对了，有一个人说对了。”请问，他们三个人中到底谁做对了？

3. 如果你有无穷多的水，一个3公升的提桶，一个5公升的提桶，两只提桶上下都不均匀，问你如何才能准确称出4公斤的水？（注意：只有这两个桶，没有第三个空桶）。

4. 只要A是红色，B就一定是绿色，B只要不是绿色，就一定是蓝色。但是，当A是红色时，C绝不会是蓝色，请判断以下3种说法，哪些是正确的，哪些是错误的？

1）只要C是蓝色，B就可能是绿色。

2）只要A不是红色，C就不可能是蓝色。

3）只要B不是绿色，A就不可能是红色。

4）只要C是蓝色，A就不可能是红色。

5. 小明住在图书馆和体育馆之间，体育馆位于图书馆和火车站之间。以下3种说法，哪些是正确的，哪些是错误的？

1）体育馆到小明住处比到火车站近。

2）小明住在体育馆与火车站之间。

3）小明的住处到体育馆比到火车站近。

编程基础考核

1. **数值类型变量a=3，变量b=7；要求在不增加变量的情况下实现变换两个变量的值。请问能不能实现？如果能，请写出代码步骤。**

a=a+b;

b=a-b;

a=a-b;

1. **链表和数组的区别在哪里？各自有什么优缺点**？

1）数组在内存中是逐个存放的，也就是说倘若数组的第一个元素在地址A,则数组第二个元素就在地址A+1。

而链表则不是，链表每个节点没有相对固定的位置关系。某个节点在地址A其后的节点不一定是A+1,而在内存的其他空闲区域，呈现一种随机的状态。

2）数组一旦显式的被申明后，其大小就固定了，不能动态进行扩充。而链表则可以，可以动态生成节点并且添加到已有的链表后面。

3）链表灵活，但是空间和时间额外耗费较大；数组大小固定，元素位置固定，但是操作不灵活，且容易浪费空间，但是时间耗费较小，尤其是元素变化不大的时候效率很高。双向链表比单向的更灵活，但是空间耗费也更大

链表的特性是在中间任意位置添加删除元素的都非常的快，不需要移动其它的元素。

链表顾名思义，要把各个元素链接起来才算撒。

通常链表每一个元素都要保存一个指向下一个元素的指针（单链表）。

双链表的化每个元素即要保存到下一个元素的指针，还要保存一个上一个元素的指针。

循环链表则把最后一个元素中保存下一个元素指针指向第一个元素。

数组是一组具有相同类型和名称的变量的集合。这些变量称为数组的元素,每个数组元素都有一个编号,

这个编号叫做下标,我们可以通过下标来区别这些元素。数组元素的个数有时也称之为数组的长度。

数组查改方便,链表增删效率高

1. **遍历循环链表时，如何判定遍历结束？**

通常对普通链表进行遍历时，一般从头节点开始，并遇到null节点时停止。而在循环链表中，当第二次到达首节点时，遍历停止

1. **列出你所知道的几种排序算法的名称。挑一个你最熟悉的排序算法讲讲大致的实现步骤。**

**冒泡排序：**比较相邻两个元素，从第一对开始比较一直到最后一对，若顺序不对就交换（感觉就像冒泡一样）。一趟比较后，最大（或最小）的会位于最后的位置，然后再以类似方式比较前面的元素。复杂度O(n²)

**选择排序：**选择排序可以说是最好理解的算法。就是每次遍历一趟，找出最小的数，放到最前端。（这里说的是最前，是指无序的队列中的最前）复杂度O(n²)

**插入排序：**遍历未排序序列。把未排序数列的第一个数和已排序数列的每一个数比较，若比它大则交换。经典的理解方式就是理解成摸牌时候理牌的顺序。我上面的实现是直接交互数字，若是把大的数直接往后移效率还会更高。复杂度O(n²)

**希尔排序：**主要就是选定一个h的有序数组来进行预排序。这样最后进行插入排序的时候，能使数据局部有序。就算交换的话，交换的次数也不会很多。这样h序列称为递增序列。希尔的性能很大部分取决于递增序列.一般来说我们使用这个序列3x + 1. 介于O(nlogn)和O(n²)之间

归并排序：核心代码是合并的函数。合并的前提是保证左右两边的数组分别有序，在合并之前和之后在Java中我们可以用断言来保证数组有序。合并的原理其实也很简单，先把a数组中的内容复制到额外储存的temp数组中去。分别用两个index指向a数组的起始位置和中间位置，保证a数组左右两边有序，比如i，j。现在开始从头扫描比较左右两个数组,若a[i]<=a[j]，则把a[i]放到temp数组中去，且i向前走一步。反正则放a[j]，且j走一步。若其中一个数组走完了，则把另一个数组剩余的数直接放到temp数组中。我们用递归的方式来实现左右两边有序。递归到数组只有1个数时肯定是有序的，再合并2个数，再退出来合并4个数，以此类推 复杂度O(nlogn).

堆排序：把数组排成二叉堆的顺序。调换根节点和最后一个节点的位置，然后对根节点进行下沉操作。时间复杂度O(nlogn)

1. **素数是只能被一和这个数本身整除（既商也没有余数）的数。查找100以内的素数，大致的算法是怎样的？**
2. **用你熟悉的编程语言，打印出9\*9乘法表。**

public static void main(String[] args) {

for (int i = 1; i <= 9; i++) {

for (int j = 1; j <= i; j++) {

System.out.print(i+"\*"+j+"="+j\*i+" ");

}

System.out.print("\n");

}

}

数据库理论基础

1. **数据库事务的四个特性是什么，各自代表什么含义？**

**原子性**:整个事务中的所有操作，要么全部完成，要么全部不完成，不可能停滞在中间某个环节。事务在执行过程中发生错误，会被回滚（Rollback）到事务开始前的状态，就像这个事务从来没有执行过一样

**一致性**:在事务开始之前和事务结束以后，数据库的完整性约束没有被破坏。

**隔离性:**隔离状态执行事务，使它们好像是系统在给定时间内执行的唯一操作。如果有两个事务，运行在相同的时间内，执行 相同的功能，事务的隔离性将确保每一事务在系统中认为只有该事务在使用系统。这种属性有时称为串行化，为了防止事务操作间的混淆，必须串行化或序列化请 求，使得在同一时间仅有一个请求用于同一数据。

**持久性:**在事务完成以后，该事务所对数据库所作的更改便持久的保存在数据库之中，并不会被回滚。

1. **drop，delete与truncate的区别**

drop直接删掉表；truncate删除表中数据，再插入时自增长id又从1开始；delete删除表中数据，可以加where字句。

1. **内连接与外连接有什么区别？外连接有哪几种类型？**

内连接指的是把表连接时表与表之间匹配的数据行查询出来，就是两张表之间数据行匹配时，要同时满足ON语句后面的条件才行。

左连接的意思是，无论是否符合ON语句后面的表连接条件都会把左边那张表的记录全部查询出来，右边的那张表只匹配符合条件的数据行。右连接则与之相反（这里同样OUTER 可以省略）。

1. **说说数据库范式种：第一范式、第二范式、第三范式的特点。**

第一范式（1NF）：强调的是列的原子性，即列不能够再分成其他几列

(无重复的列)

第二范式（2NF）：首先是 1NF，另外包含两部分内容，一是表必须有一个主键；二是没有包含在主键中的列必须完全依赖于主键，而不能只依赖于主键的一部分。

(属性完全依赖于主键[消除部分子函数依赖])

第三范式（3NF）：首先是 2NF，另外非主键列必须直接依赖于主键，不能存在传递依赖。即不能存在：非主键列 A 依赖于非主键列 B，非主键列 B 依赖于主键的情况

(属性不依赖于其它非主属性[消除传递依赖])

1. **SQL语言包括数据定义、数据操纵（Data Manipulation），数据控制（Data Control）。请问分别指的是哪些操作？**

数据定义：Create Table,Alter Table,Drop Table, Craete/Drop Index等

数据操纵：Select ,insert,update,delete,

数据控制：grant,revoke

1. **索引的作用？它的优点缺点是什么?**

索引就一种特殊的查询表，数据库的搜索引擎可以利用它加速对数据的检索。它很类似与现实生活中书的目录，不需要查询整本书内容就可以找到想要的数据。索引可以是唯一的，创建索引允许指定单个列或者是多个列。缺点是它减慢了数据录入的速度，同时也增加了数据库的尺寸大小

1. **什么叫视图？游标是什么？**

视图是一种虚拟的表，具有和物理表相同的功能。可以对视图进行增，改，查，操作，试图通常是有一个表或者多个表的行或列的子集。对视图的修改不影响基本表。它使得我们获取数据更容易，相比多表查询。

游标：是对查询出来的结果集作为一个单元来有效的处理。游标可以定在该单元中的特定行，从结果集的当前行检索一行或多行。可以对结果集当前行做修改。一般不使用游标，但是需要逐条处理数据的时候，游标显得十分重要。

1. **触发器的作用？**

触发器是一中特殊的存储过程，主要是通过事件来触发而被执行的。它可以强化约束，来维护数据的完整性和一致性，可以跟踪数据库内的操作从而不允许未经许可的更新和变化。可以联级运算。如，某表上的触发器上包含对另一个表的数据操作，而该操作又会导致该表触发器被触发