数据结构等理论

2022年6月30日 16:46

1. 大顶堆与小顶堆
   - 大顶堆：每个父节点的元素都大于等于其左右子节点的值
   - 小顶堆：每个父节点的元素都小于或等于其左右子节点的元素
2. 计算机的组成部分
   - 硬件系统 + 软件系统
   - 硬件系统
     - 控制器，存储器，运算器、输入设备、输出设备
   - 软件系统
     - 应用软件
     - 系统软件
3. 缓存的分级
   - 一级缓存：基本上都是内置在CPU的内部和CPU保持相同速度进行运行、容量小
   - 二级缓存：协调一级缓存与内存之间的工作效率
   - 三级缓存：三级缓存与二级缓存的作用类似，为读取二级缓存不够用时而设计的一种缓存手段
4. 数组和链表的区别
   - 数组在内存空间中是连续存储的，链表在内存中可以存储在任意地方
   - 数组静态分配内存，链表动态分配内存
5. 高斯分布

   横轴区间（μ-σ,μ+σ）内的面积为68.268949%。

   $$erf\left(\frac{1}{\sqrt{2}}\right) = 0.6826$$

   横轴区间（μ-2σ,μ+2σ）内的面积为95.449974%。

   - $$erf\left(\frac{2}{\sqrt{2}}\right) = 0.9544$$

   横轴区间（μ-3σ,μ+3σ）内的面积为99.730020%。

   $$erf\left(\frac{3}{\sqrt{2}}\right) = 0.9973$$

6. Hessian矩阵正定
   - 判定条件
     - 各顺序主子式的值大于0
     - 矩阵的特征值全大于0
   - 计算方法

     例：$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$ 这个四阶方阵的顺序主子式依次为：

     - $|1|、\begin{vmatrix} 1 & 2 \\ 5 & 6 \end{vmatrix}、\begin{vmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{vmatrix}、\begin{vmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{vmatrix}$

     这四个，几阶方阵就会有几个顺序主子式。

7. 矩阵的逆
   - 矩阵可逆的条件

        - 给定一个n阶方阵A，若有一个n阶方阵B，使得AB=BA=E，其中E为n阶单位矩阵，则称A可逆
        - 矩阵A可逆的充要条件时：行列式不等于0
    - 求逆的方法
        - 待定系数法
            - $\begin{bmatrix} 1 & 2 \\ -1 & -3 \end{bmatrix} \cdot \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a+2c & b+2d \\ -a-3c & -b-3d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
        - 伴随矩阵法
            - 伴随矩阵是矩阵元素所对应的代数余子式，所构成的矩阵再转置得到
            - 定理1给出了计算逆矩阵的一种方法：

              1) 计算$|A|$；　　2) 计算A^*；

              3) 写出A^{-1}，$A^{-1}=\frac{1}{|A|}A^*$.
8. 矩阵的特征值以及特征向量计算
    - 特征值及特征向量
        - 对于n阶矩阵A，如果存在数值 λ 和非0向量 α，使得 Aα = λα ，则我们称 λ 为矩阵的特征值，α 为对应 λ 的 特征向量
    - 计算方式（特征多项式）
        - 特征多项式 |λI - A| = 0
            - $\begin{pmatrix} 3 & 1 \\ 5 & -1 \end{pmatrix}$

              解：特征多项式 |λI - A| = 0为

              $\begin{vmatrix} \lambda-3 & -1 \\ -5 & \lambda+1 \end{vmatrix}$

              简化后得 (λ - 4)(λ + 2) = 0

              得矩阵得特征值为 4 和 -2

              我们将特征值4带入矩阵 (λI - A)

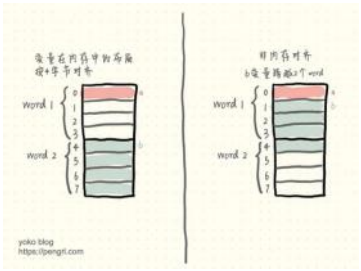
              $\begin{pmatrix} 1 & -1 \\ -5 & 5 \end{pmatrix}$

              接下来就是利用增广矩阵求解基础解系，如果忘了请查阅线性方程组章节

              得出对应得特征向量 α = (1, 1)T

              同理将特征值-2带入矩阵中，求解对应得基础解系
9. 时间复杂度和空间复杂度
    - 都用于分析算法效率
    - 时间复杂度
        - 时间复杂度衡量的是一个算法的运行速度
        - 算法中的基本操作的执行次数，为算法的时间复杂度
        - 函数执行基本操作的次数为$F(N)=N^2+2*N+10$，只保留最高阶项
    - 空间复杂度
        - 空间复杂度衡量算法在运行过程中所占用的内存空间的大小
        - 空间复杂度不是程序占用了多少空间，空间复杂度算的是变量的个数

10. 贪心算法和动态规划的区别
    - 贪心算法
        - 每次只考虑当前，不能保证得到全局最优解，只保证当前最优解，一般时间复杂度较低
    - 动态规划
        - 把一个大的问题进行拆分，细分成一个个小的子问题，且能够从这些小的子问题的解推导出原问题的解
        - 具有最优子结构性质和重叠子问题性质，可以得到全局最优解。
        - 每步所做的选择往往依赖于相关子问题的解，因而只有在解出相关子问题时才能做出选择
11. 动态规划和递归的区别
    - 相同点
        - 两者都采用了分而治之的思想
        - 将原始问题分解为一个个小问题进行处理
    - 区别
        - 递归
            - 自上而下的思想
            - 从大问题到小问题
            - 递归可能存在重复运算
        - 动态规划
            - 自下而上的思想
            - 先解决小问题，由小问题的解得到大问题的解
            - 动态规划会储存每个小问题的结果，从而它的计算速度会比递归要快
12. 优先级队列
    - 概念
        - 一种特殊的队列，在优先队列中，元素被赋予优先级，当访问队列元素时，具有最高优先级的元素最先被删除
        - 优先队列按照元素的优先级决定出队顺序：优先级高的元素先出队，优先级低的元素后出队
    - 适用场景
        - 任务调度：根据优先级执行系统任务
        - 选择问题：查找第 k 个最小元素
    - 实现方法
        - 大顶堆、小顶堆
13. 深度优先与广度优先的区别
    - 深度优先
        - 深度优先采用栈
        - 深度优先对每一个可能的分支路径深入到不能再深入为止
        - 深度优先有回溯操作，不保留全部结点，占用空间少
    - 广度优先
        - 广度优先采用队列
        - 广度优先层次遍历，从上往下对每一层依次访问，在每一层中，从左往右访问结点
        - 广度优先保留全部结点，占用空间大
14. for循换和while循换的区别
    - for循换
        - 循换次数已知
        - 循换变量只在循环体内有效
    - while循换
        - 循换次数未知
        - 先判断条件再执行循环体
        - 循环变量出了循换还能继续使用
15. 内存对齐

- CPU访问内存时，并不是逐个字节访问，而是以字长（word size）为单位访问
  - 比如32位的CPU，字长为4字节，那么CPU访问内存的单位也是4字节。
- 内存对齐目的
  - 减少CPU访问内存的次数
    - 读取8个字节的数据，一次读取4个字节那么只需要读取2次
    - 

16. Dijkstra算法（迪杰斯特拉算法）
    - 作用
      - 解决的是有权图中最短路径问题，给定图G和起点，得到S到达其他每个顶点的最短距离
    - 思想
      - 每次找到离起点最近的点，然后用它到起点的最短路来更新其他的点到起点的最短路（每个点只用一次）
17. Prim算法（普里姆算法）
    - 作用
      - 在加权连通图中搜索最小生成树
      - 根据图中权值找到连接所有顶点的最短路径
    - 思想
      - 选取权值最小边的其中一个顶点作为起始点。
      - 找到离当前顶点权值最小的边，并记录该顶点为已选择。
      - 重复第二步，直到找到所有顶点，就找到了图的最小生成树
18. **********
19. 数据库事务
    - 定义
      - 由一系列的数据库操作组成
      - 要么全部执行，要么全部不执行
      - 是一个不可分割的工作单位
    - 四大特性
      - 原子性
        - 事务包含的所有操作要么全部成功，要么全部失败回滚
      - 一致性
        - 事务必须使数据库从一个一致性状态变换到另一个一致性状态
      - 隔离性
        - 多个并发事务之间要相互隔离
      - 持久性
        - 事务一旦被提交了，那么对数据库中的数据的改变就是永久性的
    - 读取数据存在的问题
      - 脏读
        - 一个事务读取了另一个未提交的事务中的数据
        - 修改时允许读取
      - 不可重复读
        - 对于数据库中的某个数据，一个事务多次查询却返回了不同的数据值
      - 虚读
        - 事务非独立执行时发生的一种现象
        - 读取时允许插入

- 事务隔离级别
  - 串行化
    - 一个一个排队执行
    - 可避免 脏读，虚读，不可重复读
  - 可重复读
    - 当一个事务启动后，不再允许进行修改操作
    - 可避免不可重复读
  - 读提交
    - 只能读到已经提交了的内容
    - 可避免脏读
  - 读未提交
    - 可以读到未提交的内容
    - 可能会产生脏读、不可重复读、幻读

20. 数据库连接方式
    - 内连接
      - 用于查询来自两个或多个相关表的数据
    - 左连接
      - 用于查询来自多个表的数据
    - 右连接
      - 组合来自两个或多个表的数据
    - 交叉连接
      - 连接两个或多个不相关的表
21. 数据库结构
    - 客服端，服务端，表，字段，记录，单元格
22. 数据库操作
    - 数据库以及数据库中的表
      - CREATE
        -
```
-- CREATE TABLE <表名>
-- (
--         <列名1> <数据类型> <约束>,
--         <列名1> <数据类型> <约束>,
--         ...
--         <表约束1>,
--         <表约束2>,
--         ...
-- )
```
      - DROP
        - DROP TABLE <表名>
      - ALERT
        - ALTER TABLE <表名> ADD <列名> <类型>;
        - ALTER TABLE <表名> DROP COLUMN <列名>;
    - 记录
      - SELECT
      - ISERTE
      - UPDATE
      - DELETE
    - 数据库删除方法
      - Delete
        - 删除数据表中的行
        - 可以删除某一行，也可以在不删除数据表的情况下删除所有行
      - Truncate
        - 删除数据表中的数据
        - 仅数据表中的数据，不删除表

        - Drop
            - 删除数据表或数据库，或删除数据表字段
23. 数据库索引的作用
    - 可以快速访问数据库表中的特定信息。
    - 提高数据的检索速度
    - 加快表与表之间的连接速度
24. 数据库范式
    - 目的
        - 解决关系数据库中数据冗余、更新异常、插入异常、删除异常问题
    - 1NF
        - 强调属性的原子性约束
        - 要求属性具有原子性，不可再分解
    - 2NF
        - 强调记录的唯一性约束
        - 数据表必须有一个主键
        - 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
    - 3NF
        - 强调数据属性冗余性的约束
        - 非主键列必须直接依赖于主键
        - 消除了非主属性对码的传递函数依赖
    - BCNF
        - 防止主键的某一列会依赖于主键的其他列
    - 4NF
        - 非主属性不应该有多值
        - 限制关系模式的属性间不允许有非平凡且非函数依赖的多值依赖
    - 5NF
        - 消除了4NF中的连接依赖
25. 1列不可再分 2非主键字段必须依赖主键字段 3非主键字段不依赖其他非主键字段