快指针先走,慢指针与快指针间隔K的距离再开始同时移动,当快指针移动到尾部后,慢指针就是倒数第K个节点 删除倒数第K个节点 快指针每次走两步,慢指针每次走一步,如果相遇则是环形链表,如果有尽头,就不是环形链表 双指针法 判断链表是否是环形链表 单向链表 先判断两个链表的长度,设长度排列大小为 a 、b (a) b),让最长的先移动,移动的步子为a-b。移动结束后,两个链表上的指针同时移动,对比每一个节点。若都是相等则链表相交 判断两个链表是否相交 链表 堆栈法 翻转链表 原地翻转 双向链表 概念:一棵空树或它的左右两个子树的高度差的绝对值不超过1 自平衡二叉查找树 平衡二叉树 红黑树 数据结构 二叉树的最大深度 一些递归解法 两个二叉树是否相等 二叉树 前序、中序、后序遍历 前序遍历 中序遍历 后序遍历 深度搜索 层序遍历 遍历 搜索最短路径 计算机科学 用堆栈实现队列 队列 用队列实现堆栈 两两比较 冒泡排序 从无序的集合中,选择最大或最小的值出来,然后再从剩下的无序集合中选择下一个最大或最小值,直到排序完无序的集合。 选择排序 思路是分治法, 子序列有序后, 再将有序的队列归并 选择值后比较,分为大于和小于两部分,然后分别对这两部分进行选值再排序,直到排序完成 十大排序算法 堆排序 计数排序 算法 希尔排序 分治算法 贪心算法 四大常用算法 动态规划 A\*算法 计算机组成原理 操作系统 编译原理 编程语言 JavaScript TypeScript 软件开发 逆矩阵 行列互换 转置矩阵 正交矩阵 矩阵乘以自己的转置矩阵等于单位矩阵 运算 四元数 存在意义 面积法 计算机数学 判断点是否在多边形内 射线法 圆形与矩形 分离轴定理,多边形做每条边的法向量,把两个多边形在法向量上投影,如存在不重叠部分,那就是未碰撞 凸多边形与凸多边形 2D 方面 凹多边形需要分割为凸多边形 多边形与多边形碰撞 几何计算 只判断部分, 不必对所有对象进行检测 点与线的距离 3D方面 两个具有速度的球体运动, 是否会碰撞 把JS方法绑定到C++中去 JSBing 热更新 CocosCreator lo calS to rage H5方向 定制JS引擎 游戏编程 渲染流 LayaBox U n ity3D 手游、端游 Unrea14 2D 物理引擎 Box2D 游戏引擎 物理引擎 3D 物理引擎 W ebGL 0 penGLES 0 penGL DirectX 从硬盘从加载数据到内存中, 再从内存加载到显存中 应用阶段 调用D raw Call 顶点着色器 曲面细分着色器 几何着色器 几何阶段 计算机图形学 图形渲染 渲染管线 屏幕映射 三角形设置 三角形遍历 片元着色器 光栅化阶段 逐片元操作 模板测试 混合 深度测试 游戏音效 游戏物理与动画 为AI设定多个状态行为, 通常以某些事件去触发状态的转变 结构与状态机不一样, 行为树可以做更多的组合: 选择执行、顺序执行、同步执行 行为树 '一些"思考",对周围环境进行评估再决策。 其他 博弈树 剪枝算法 极小极大值算法 遗传算法 神经网络 建立TCP连接才能通信,而HTTP是无状态协议,后来引入了Cookie НТТР 是对HTTP的加密版本 HTTPS 应用层 W ebSocket 是对HTTP的一次升级,保证一个长连接 文件传输协议 表示层 七层结构 会话层 可靠有序、重传机制是全部重传

游戏客户端

网络

日常开发问题

网络安全 防止图集来源差异 降低D raw Call 防止交叉渲染 防止Label打断渲染 CPU 使用对象池 减少垃圾回收 耗时逻辑,分帧处理 使用对象池,减少频繁创建与销毁 游戏优化 图集打包,减少散图所占大小 GPU 压缩图片尺寸 资源加载与卸载的时机选择 无损压缩图片资源 尽可能少把ttf字体文件一起打包,缩小游戏包体 主要逻辑在服务器,客户端只做输入与显示 比较好防范外挂,因为服务器的状态才是准确的,可以直接判断到哪些请求是恶意的。 状态同步 比较好做断线重连, 因为服务器可以选择存储游戏的状态 主要逻辑在客户端, 服务器只转发操作指令 不好防外挂, 缺少一个仲裁的服务器 同步 不好做断线重连 防外挂的手段 复盘检验,将指令进行模拟检验,看结果是否一样 一般需要从游戏开始的第一帧开始,服务器需要把第0帧到当前帧的所有操作数据发

Crash后重连

非Crash重连

服务器需要下发客户端当前帧到服务器当前帧内的所有操作指令

重连的方式

为什么是三次

为什么是四次

面向报文, 无连接, 不可靠的传输层协议, 主要追求一个快

实现一层可靠的机制, 选择重传、保证传输速度

双向通信

三次握手

TCP

UDP

传输层

网络层

物理层

数据链路层