

师弟C++学习计划

一、C++语法

书籍推荐看C++PrimerPlus或C++Primer其中一本

1. C++基础

- 变量与类型
 - 变量的声明和初始化
 - 基本数据类型 (如int, float, double, char, bool)
 - 枚举类型
 - 类型转换
- 字符串、向量和数组
 - 字符串操作 (std::string)
 - C风格字符串 (char数组)
 - 数组的声明与使用
 - 向量的使用 (std::vector)
- 表达式
 - 算术表达式
 - 逻辑表达式
 - 关系表达式
 - 条件表达式
 - 运算符优先级
- 语句
 - 控制语句 (if, else, switch)
 - 循环语句 (for, while, do-while)
 - 跳转语句 (break, continue, return, goto)
- 函数
 - 函数的定义与调用
 - 参数传递 (值传递、引用传递、指针传递)
 - 函数重载
 - 内联函数
 - 函数模板
- 类
 - 类的定义与对象的创建
 - 构造函数与析构函数
 - 访问控制 (public, private, protected)
 - 成员函数和成员变量
 - this指针
 - 继承与多态
 - 运算符重载
 - 友元函数和友元类

2. C++ STL标准库

- **IO库**
 - 输入输出流 (cin, cout, cerr)
 - 文件输入输出 (ifstream, ofstream, fstream)
 - 字符串流 (stringstream)
- **顺序容器**
 - vector
 - deque
 - list
 - array
- **关联容器**
 - set
 - multiset
 - map
 - multimap
 - unordered_set
 - unordered_map
- **泛型算法**
 - 查找算法
 - 排序算法
 - 拷贝与替换算法
 - 数值算法 (如accumulate, inner_product, partial_sum, adjacent_difference)
- **动态内存**
 - new和delete操作符
 - 智能指针 (std::unique_ptr, std::shared_ptr, std::weak_ptr)
 - 动态数组 (std::vector)

二、算法学习

1. 算法

- **算法复杂度**
 - 时间复杂度 (Big O notation)
 - 空间复杂度
 - 最坏情况、平均情况、最优情况分析
- **递归与迭代**
 - 递归的基本概念
 - 递归与迭代的比较
 - 递归问题的解决方法 (如汉诺塔、斐波那契数列)

2. 线性表

- **数组**
 - 一维数组
 - 多维数组
 - 动态数组
- **链表**

- 单链表
- 双向链表
- 循环链表
- 链表的基本操作（插入、删除、查找、反转）
- **栈**
 - 栈的基本操作（入栈、出栈、取栈顶元素）
 - 栈的应用（如括号匹配、中缀表达式转后缀表达式）
- **队列**
 - 队列的基本操作（入队、出队、取队首元素）
 - 循环队列
 - 双端队列

3. 树

- **树的基本概念**
 - 树的定义
 - 树的性质
 - 树的表示方法（如孩子表示法、双亲表示法）
- **二叉树**
 - 二叉树的定义和性质
 - 二叉树的遍历（前序、中序、后序、层序遍历）
 - 二叉搜索树（BST）
 - 平衡二叉树（如AVL树）
- **高级树结构**
 - 堆（最大堆、最小堆）
 - B树、B+树
 - 红黑树
 - 字典树（Trie）

4. 图

- **图的基本概念**
 - 图的定义和表示方法（邻接矩阵、邻接表）
 - 图的性质（有向图、无向图、连通图、完全图）
- **图的遍历**
 - 深度优先搜索（DFS）
 - 广度优先搜索（BFS）
- **最短路径算法**
 - Dijkstra算法
 - Floyd-Warshall算法
 - Bellman-Ford算法
- **最小生成树**
 - Prim算法
 - Kruskal算法
- **拓扑排序**
 - 拓扑排序的概念和实现方法

5. 查找

- **顺序查找**
 - 基本顺序查找算法
- **二分查找**
 - 二分查找的实现和分析
 - 二分查找的变形问题
- **哈希查找**
 - 哈希表的基本概念
 - 哈希函数
 - 冲突处理（开放地址法、链地址法）

6. 排序

- **交换排序**
 - 冒泡排序
 - 快速排序
- **选择排序**
 - 简单选择排序
 - 堆排序
- **插入排序**
 - 直接插入排序
 - 希尔排序
- **归并排序**
 - 二路归并排序
 - 多路归并排序