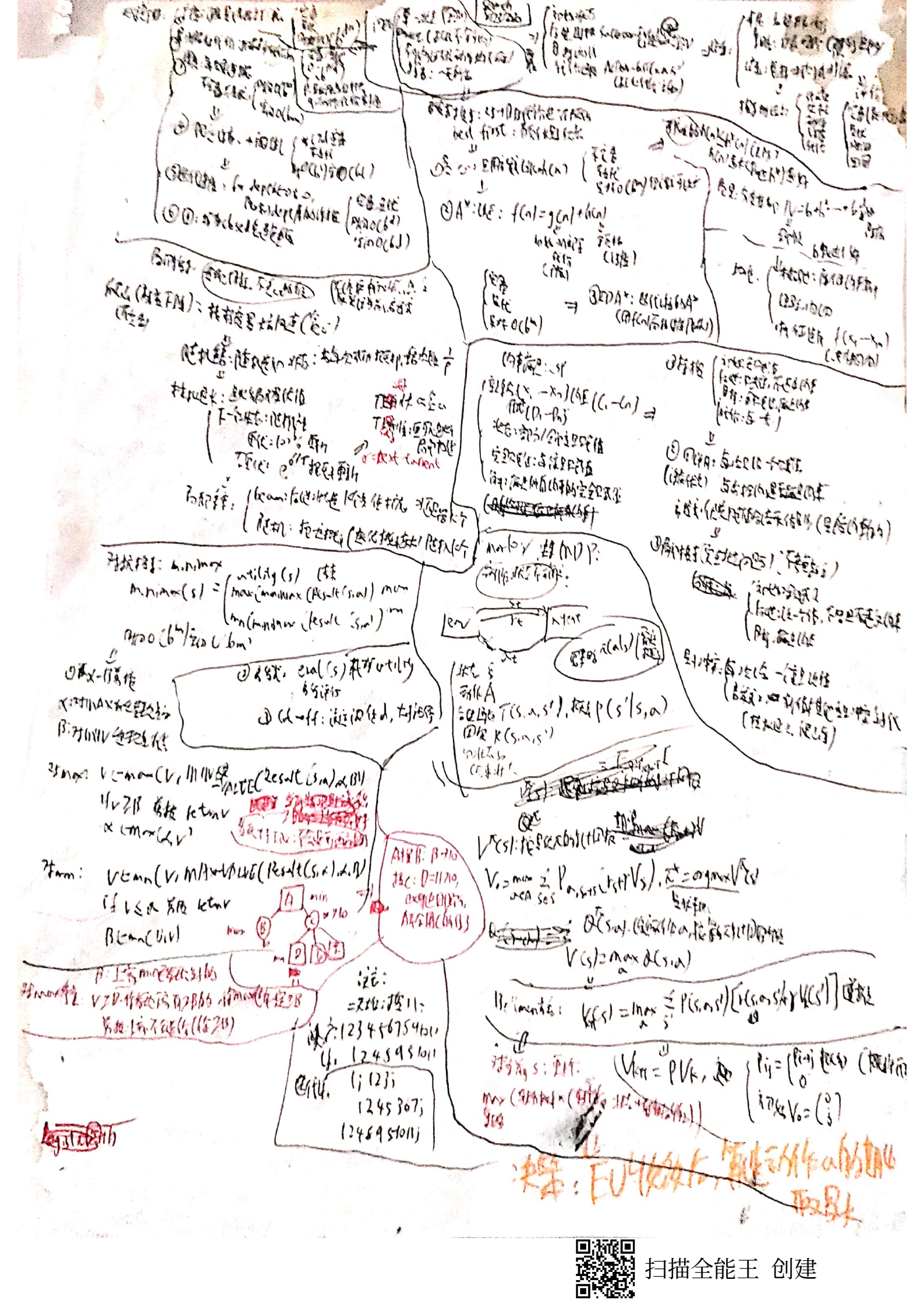
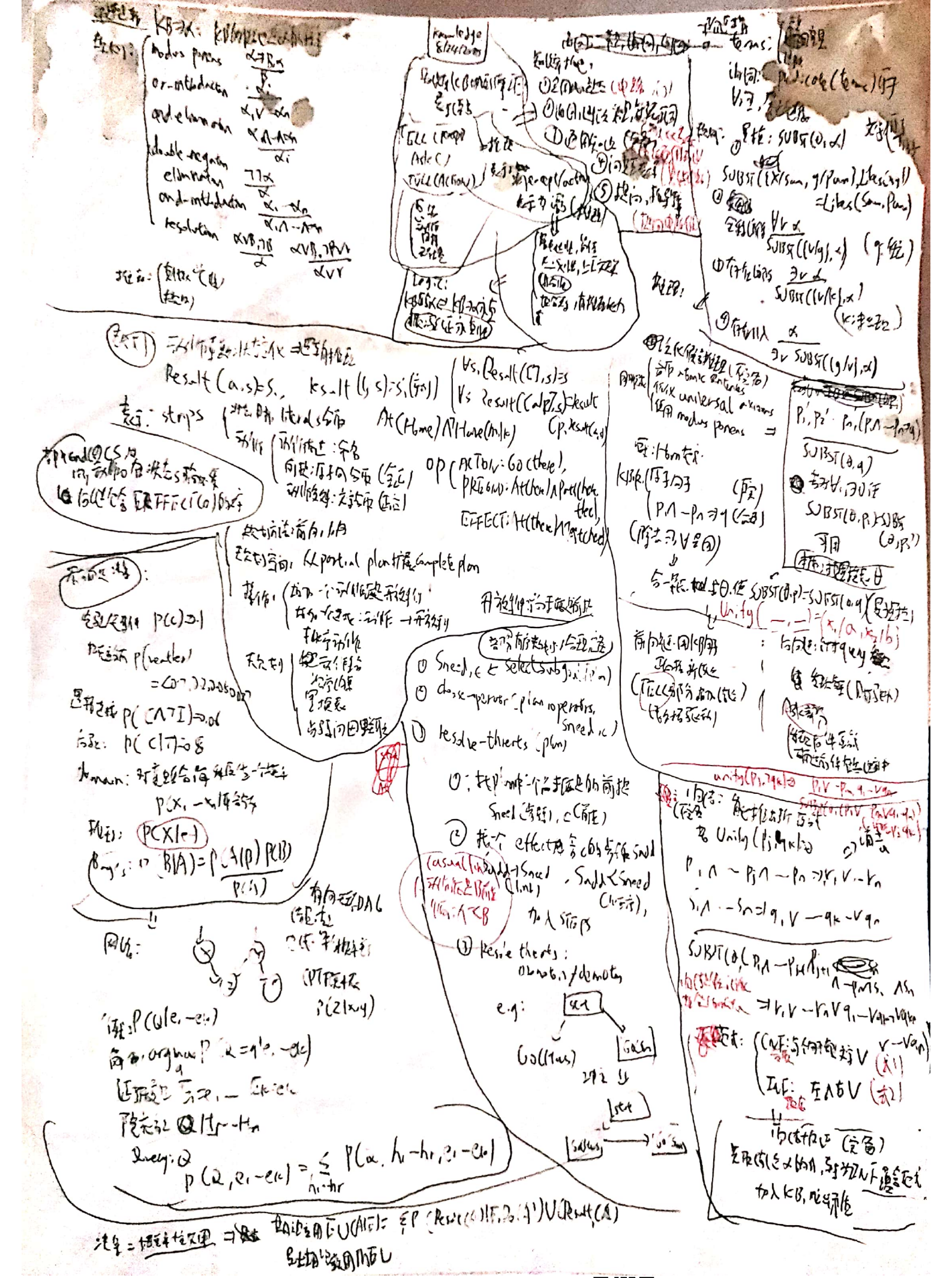
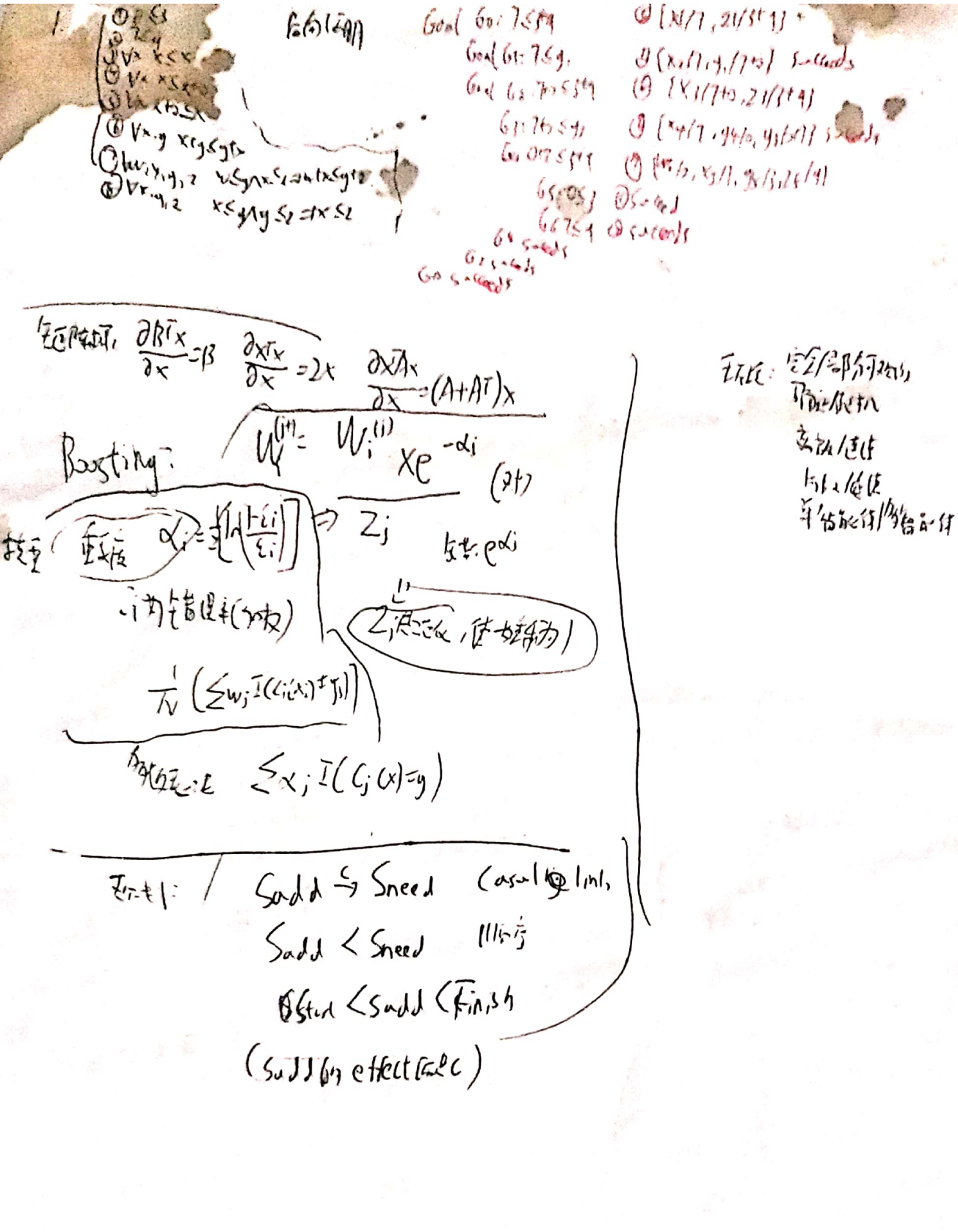
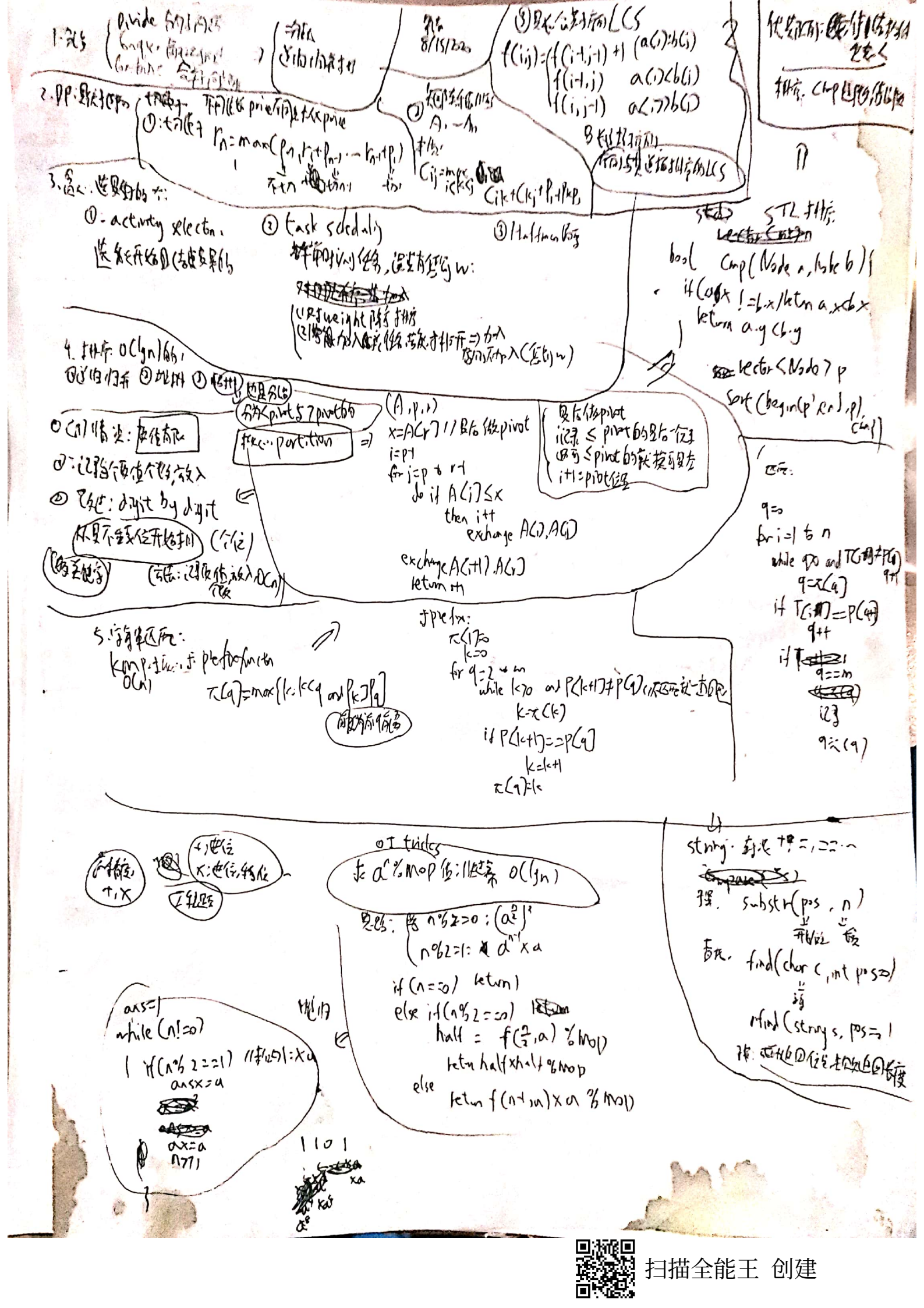






扫描全能王 创建


8/15/2020
STL 排序
扫描全能王 创建

线性：① 数组 a[100] a[100][100]
② 变长数组：vector .size() push_back() pop_back()

③ 队列：先进先出 push() 入队 pop() 出队 front() 头 back() 尾

数据结构 8/14/2020

④ 栈：后进先出 top() pop() push() empty()

应用：括号匹配，表达式求值

⑤ 线性表

树形：二叉树
遍历：先：根左右 中：左根右 后：左右根
中+先/后确定一个二叉树

二叉排序树：左<根<右

哈夫曼树(Huffman)

完全二叉树 根 左右

查找：二分
low=1
high=L
while(low<=high)
mid=(low+high)/2
if(==) return mid
else if (kval < a(mid)) high=mid-1
else low=mid+1

平衡二叉树 AVL O(lgn)
B树

map<int string> personnel
① insert(1,"one")
② [1]="one"
查找：map student.find(1)
删除：erase(1)

set<int> a
st.insert(x)
st.pop_back(x)
st.erase
st.find(x)

hash表 ⇒ hash_map<string int>
string hash["..."]=123

图：① 最短路：dijkstra
② 最小生成树：prim

优先队列
定义：struct cmp {
friend bool operator ()(int x, int y)
{
return x>y; // x 优先级低
}
};
priority_queue<int, vector<int>, cmp> q
push() 插入并排序
pop() 删除
top() 队首
empty() . size()