

网络围棋对弈与教学平台 软件系统设计

目 录

1. 系统分布设计	D-NGGTP-01-01
2. 层次示意图	D-NGGTP-01-02
3. 数据结构设计	D-NGGTP-01-03
4. 算法设计	D-NGGTP-01-04

1. 系统分布设计（负责人：全体）

1) 表示层服务:

用户信息展示服务

游戏房间信息展示服务

棋盘信息显示服务

聊天信息显示服务

道具信息展示服务

弈币信息展示服务

棋谱信息公众号服务

评价信息公众号服务

问题信息公众号服务

围棋资讯公众号服务

2) 应用逻辑层服务:

用户信息的增删改查服务

游戏房间状态信息的增删改查服务

棋盘信息的增删改查服务

聊天信息的增删改查服务

弈币余额的增删改查(CRUD)服务

道具库存的增删改查服务

棋谱信息存储、删除、修改、查询服务

围棋资讯存储、删除、修改、查询、发布服务

评价信息存储、删除、修改、查询服务

问题信息存储、删除、修改、查询服务

3) 数据存储服务

用户信息的数据库操作服务

游戏房间状态信息的数据库操作服务

棋盘信息的数据库操作服务

聊天信息的数据库操作服务

弈币余额的数据库操作服务

道具库存的数据库操作服务

围棋资讯的数据库操作服务

用户信息的数据库操作服务

弈币信息的数据库操作服务

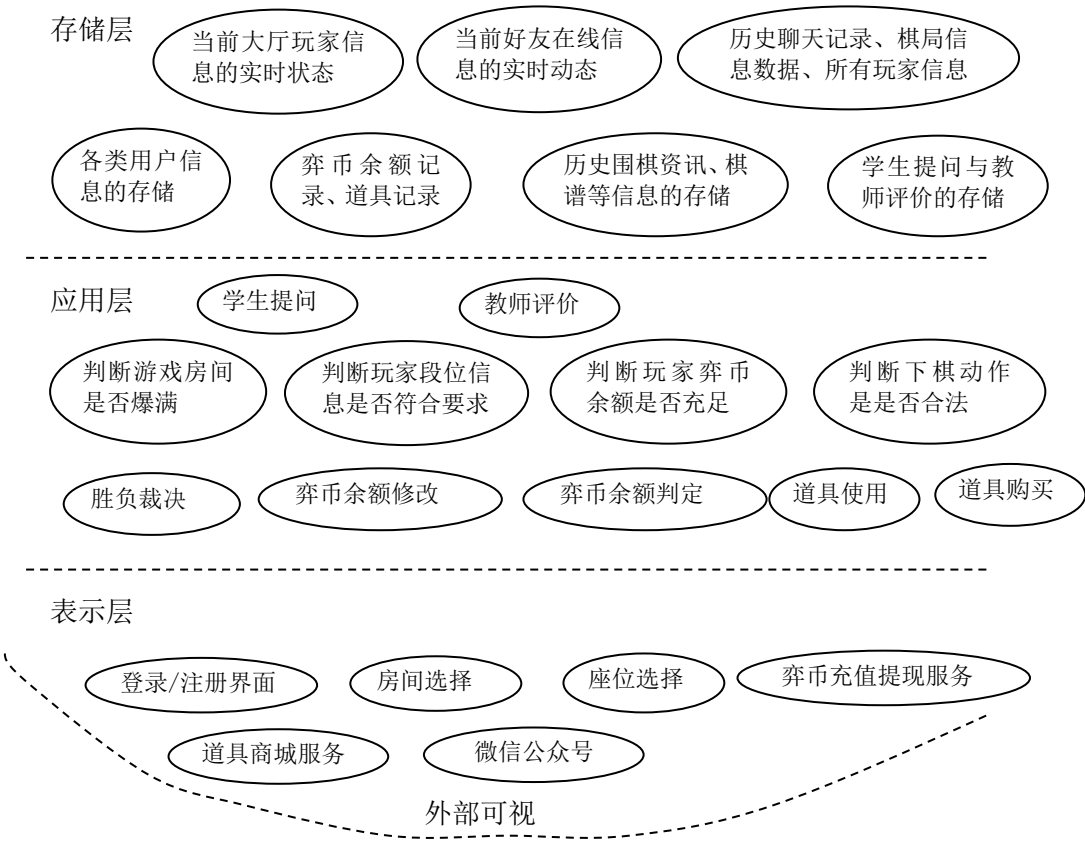
棋谱信息的数据库操作服务

评价信息的数据库操作服务

问题信息的数据库操作服务

围棋资讯的数据库操作服务

2. 层次示意图（负责人：全体）



3. 数据结构设计

1) 用户信息 (负责人: 周智)

说明 \ 字段	名称	类型	长度	处理说明
用户标识	ID	int	15	用户类型+日期+序号
用户类型	type	int	10	老师或者学生
用户昵称	nick_name	string	10	汉字、英文、基础符号
真实姓名	real_name	string	10	汉字、英文
手机号	phone_number	string	15	11 位数字
邮箱	email	string	15	英文、数字、基础符号
年龄	age	int	5	3 位以下数字
身份证号	ID_number	string	20	任意字符
真实段位	rank	int	5	2 位以下数字
游戏积分	point	int	10	任意数字
游戏道具数量	tools_number	int	10	任意数字
弈币金额	currency	int	10	任意数字
游戏记录	R_ID	int	30	用户 ID+游戏开始时间

2) 游戏大厅 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
个人信息记录	person_info	用户(类名)	15	同用户类
特定用户的好友列表	friend_list	String	变长	通过处理该字符串, 来对好友进行增减操作
房间信息	rooms_info	int[][]	100	每一个房间有唯一的 ID, 记录在 int 数组第一维, 第二维记录该房间的空满情况。

3) 游戏房间 (负责人: 聂宇翔)

字 段 说明	名称	类型	长度	处理说明
房间 ID	room_ID	int	4	不可更改, 每一个房间有唯一的 ID 进行标识
房间在线人数	room_online_number	int	4	实时显示当前房间的在线人数
房间名称	room_name	char[]	100	该房间的名称, 不可修改
房间类型	room_type	int	4	房间类型包括对弈房间 (用 0~2 进行标记) 和教学房间 (用 3 进行标记)
服务器负载状态	server_status	int	4	主要包括 4 种服务器负载状态: 0~3, 数值越大, 当前在线人数越多
世界广播信息栏	world_message	CWnd	100	承载当前的世界广播
区内玩家列表	room_players	String[]	变长	每一个 String 记录一个玩家的简略信息
座位详细信息	chair_info	String	变长	记录各个座位的信息, 放在一个 String 中进行管理

4) 下棋记录 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
棋手 ID	player_ID	int	15	即该棋手的用户 ID
棋手执子颜色	player_color	bool	100	0: 白 1: 黑
下棋位置	player_pos	int	4	每一个位置有唯一的编号
下棋时间	player_time	int	4	时间的从 1970 年 1 月 1 日 0 时开始计时的总长度, 单位为 s
当前棋盘状态	go_status	String	变长	用变长 String 存放当前棋盘状态

5) 聊天记录 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
学生 / 棋手 1 ID	player1_ID	int	4	相应用户的用户 ID
教师 / 棋手 2 ID	player2_area	int	4	相应用户的用户 ID
聊天记录时间	chatting_time	int	4	时间的从 1970 年 1 月 1 日 0 时开始计时的总长度, 单位为 s
类型	type	int	4	0: 使用文字 1: 使用语音 2: 使用道具
记录内容	chatting_content	String	变长	视使用的聊天工具类型而定

文档编号: D-NGGTP-01

设计: 聂宇翔

文档名称: 网络围棋对弈与教学平台

审查: 张奕驰

6) 道具增减记录 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
道具 ID	tool_ID	int	4	每一个道具都有唯一的一个 ID
增加/减少	change	bool	1	0: 增加 1: 减少
数量	number	int	4	道具增加或减少的数量
时间	time	int	4	时间的从 1970 年 1 月 1 日 0 时开始计时的总长度, 单位为 s

7) 魅力值变更记录 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
教师 ID	teacher_ID	int	4	教师玩家的唯一用户 ID
魅力值变更量	number	int	4	魅力值增加或减少的数量
魅力值增加/减少	change	bool	1	0: 增加 1: 减少
变更时间	time	int	4	时间的从 1970 年 1 月 1 日 0 时开始计时的总长度, 单位为 s

8) 积分变更记录 (负责人: 聂宇翔)

说明 \ 字段	名称	类型	长度	处理说明
增加/减少	change	bool	1	0: 增加 1: 减少
棋手 1 ID	player1_ID	int	4	相应用户的用户 ID
棋手 2 ID	player2_ID	int	4	相应用户的用户 ID
数量	number	int	4	积分增加或减少的数量

9) 用户道具库 (负责人: 张奕驰)

说明 \ 字段	名称	类型	长度	处理说明
用户 ID	userID	int	4	用户 ID
道具 ID	itemID	int	4	道具 ID
道具堆叠数量	stack	int	4	道具堆叠数量
道具属性位	flag	int	4	从 LSB 往 MSB 方向, 分别为: 不可交易位、冻结位、绑定位。剩余预留待定

10) 道具 (负责人: 张奕驰)

说明 \ 字段	名称	类型	长度	处理说明
道具 ID	itemID	int	4	道具 ID
道具类别	category	int	4	用于展示于商城不同标签页中
道具价格	price	int	4	商城中道具的价格
是否在商城中上架	showInShop	bool	1	是否在商城中上架

11) 弈币余额 (负责人: 张奕驰)

说明 \ 字段	名称	类型	长度	处理说明
用户 ID	userID	int	4	
弈币余额	coin	int	4	直接填数就行了

12) 弈币交易记录 (负责人: 张奕驰)

说明 \ 字段	名称	类型	长度	处理说明
用户 ID	userID	int	4	用户 ID
交易发生时间 (带 timezone)	timestamp	datetime-tz	8	交易发生时间 (带 timezone)
数额	amount	int	4	数额
对方 / 平台方 名称	peer	varchar	-	对方 / 平台方名称
交易 ID (支付 平台的交易单 号)	transID	int	4	交易 ID (支付平台的 交易单号)

13) 辅导课程记录 (负责人: 张嘉琦)

说明 \ 字段	名称	类型	长度	处理说明
课程编号	courseID	int	4	课程编号
教师 ID	teacherID	int	4	授课教师编号
时间	time	datetime-tz	8	授课时间
课程类型	type	char	1	对应普通课程或技 巧课程
学生 ID	studentID	int[]	-	上课学生的编号
弈币	coin	int	4	消耗弈币数量

文档编号: D-NGGTP-01

设计: 聂宇翔

文档名称: 网络围棋对弈与教学平台

审查: 张奕驰

14) 问题记录 (负责人: 张嘉琦)

字 段 说明	名称	类型	长度	处理说明
学生编号	userID	int	4	用户编号
问题编号	questionID	int	4	问题编号
提问时间	timeQuestion	datetime-tz	8	提问时间
问题内容	question	vchar	-	问题内容
回答时间	timeAnswer	datetime-tz	8	回答时间
回答内容	answer	vchar	-	回答内容

15) 评价记录 (负责人: 张嘉琦)

字 段 说明	名称	类型	长度	处理说明
评价编号	evaluationID	int	4	评价编号
教师编号	teacherID	int	4	教师编号
学生编号	studentID	int	4	学生编号
评价时间	time	datetime-tz	8	评价时间
评价内容	evaluation	vchar	-	评价内容

16) 消息记录 (负责人: 张嘉琦)

字 段 说明	名称	类型	长度	处理说明
消息编号	messegeID	int	4	用户 ID
消息类型	type	char	1	消息所属类型, 分为 资讯、棋谱、规则和 教学四类
发布时间	time	datetime-tz	8	发布时间

4. 算法设计

1) 记录棋盘中的“块”算法设计 (负责人: 聂宇翔)

说明: 棋盘大小为 19×19 , 现用 `map[19][19]` 数组存放当前的棋局

输入: 棋盘的位置 `[i, j]`

输出: 无

结果: 记录以位置 `[i, j]` 为开始的块至 `block` 数组中, 记录块的大小至 `blockLength` 变量中。

思路: 递归求解, 每搜索到一个位置时, 分别判断该位置上下左右的棋子是否与该位置的棋子颜色相同, 若相同, 则记录该邻居的位置信息, 并递归进行块搜索。

其中 “`isInBlock`” 用于计算该位置是否已经搜索过了, 避免无限递归。

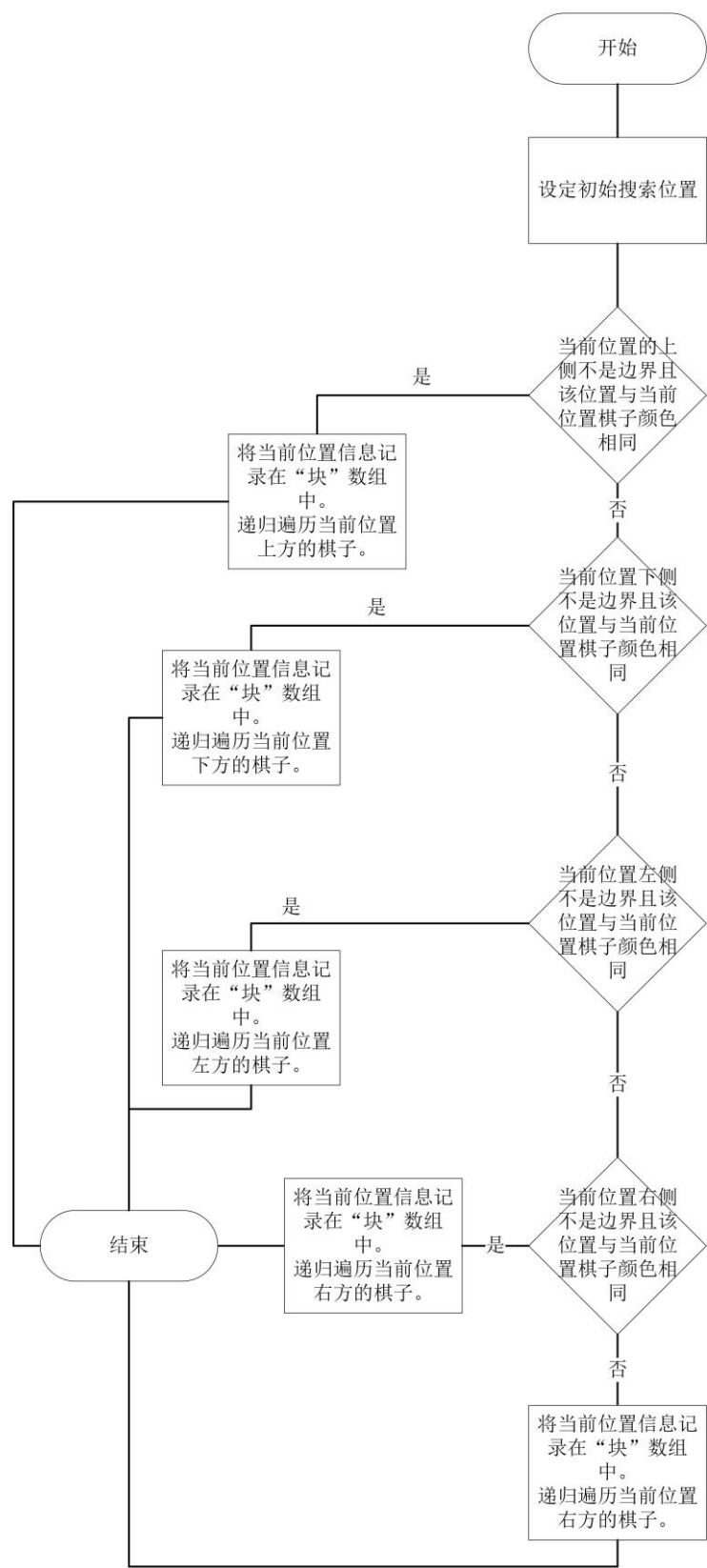


图 4-1 记录棋盘中的“块”算法流程图（负责人：聂宇翔）

2) 判断当前块是否有“气”的算法设计 (负责人: 聂宇翔)

当通过 1) 中算法已经得知一个确定的“块”之后, 我们可以对这个块是否有“气”进行判断:

输入: 无

输出:

true: 该块有“气”

false: 该块无“气”

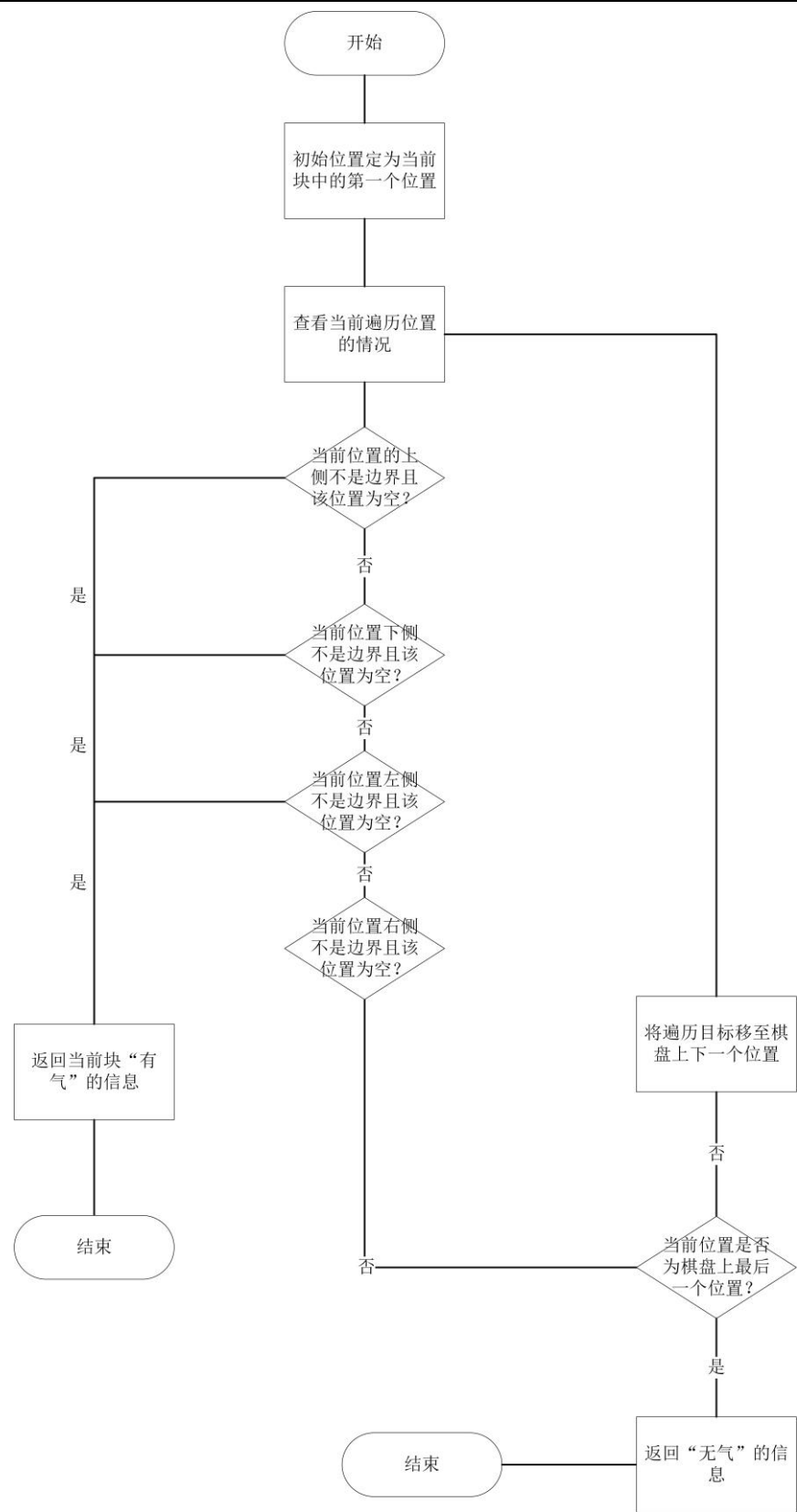


图 4-2 判断当前块是否有“气”的算法流程图

3) 吃子算法设计 (负责人: 聂宇翔)

在已有 1) 和 2) 算法的基础上, 我们可以对“吃子”的算法进行设计。

“吃子”的基本单位是“块”, 如果一个块没有“气”了, 那么它就可以被吃掉。

输入: 无

输出: 无

效果: 将当前棋局中被吃掉的“块”的位置清零。

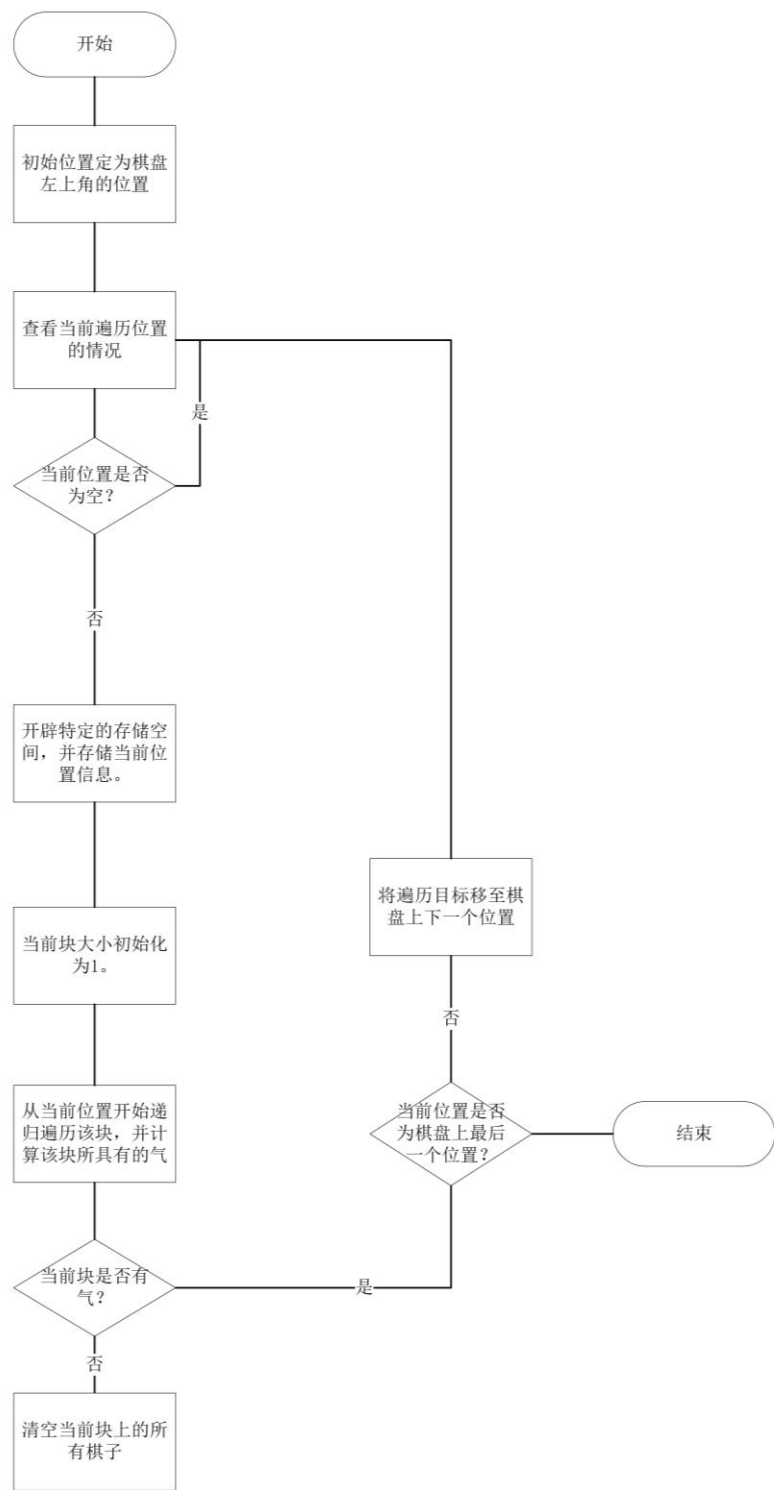


图 4-3 吃子算法流程图（负责人：聂宇翔）

4) 胜负判定算法设计 (负责人: 聂宇翔)

注: 本处的胜负判定单指正常情况下的胜负判定, 不包括意外终止棋局后的胜负判断。

本处的胜负判定采用“数子法”。

输入: 当前棋盘信息

输出:

0: 执白子者赢

1: 执黑子者赢

2: 平局

算法详情:

1. `black_score` = 棋盘上剩余黑子的数量

2. `white_score` = 棋盘上剩余白子的数量

3. `black_score` += 黑子围成的空格总数量。

4. `white_score` += 白子围成的空格总数量。

返回:

当 `white_score > black_score` 时, 返回 0

当 `white_score < black_score` 时, 返回 1

当 `white_score == black_score` 时, 返回 2

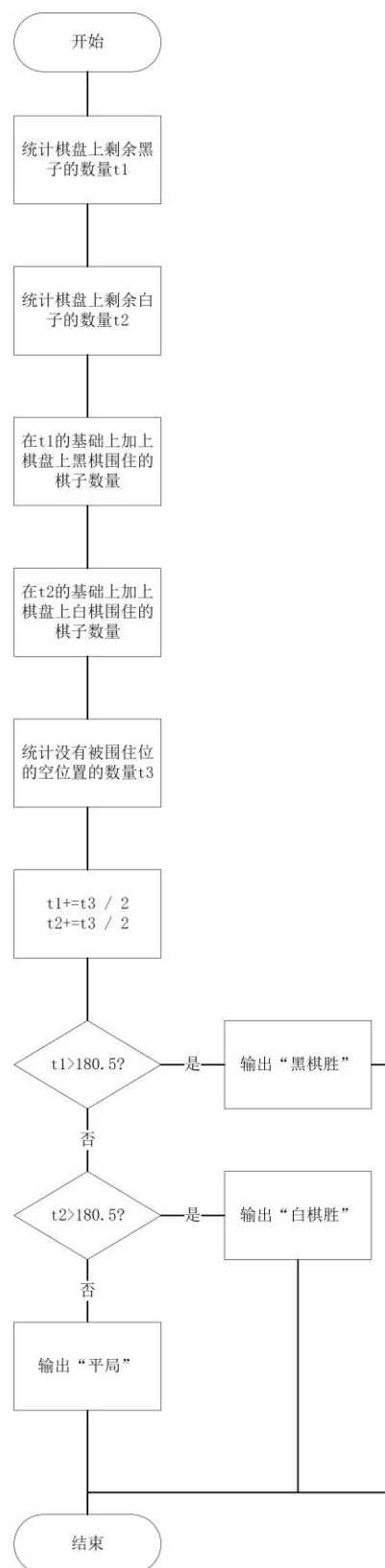


图 4-4 胜负判定算法流程图（负责人: 聂宇翔）

5) 弈币购买道具算法设计（负责人：张奕驰）

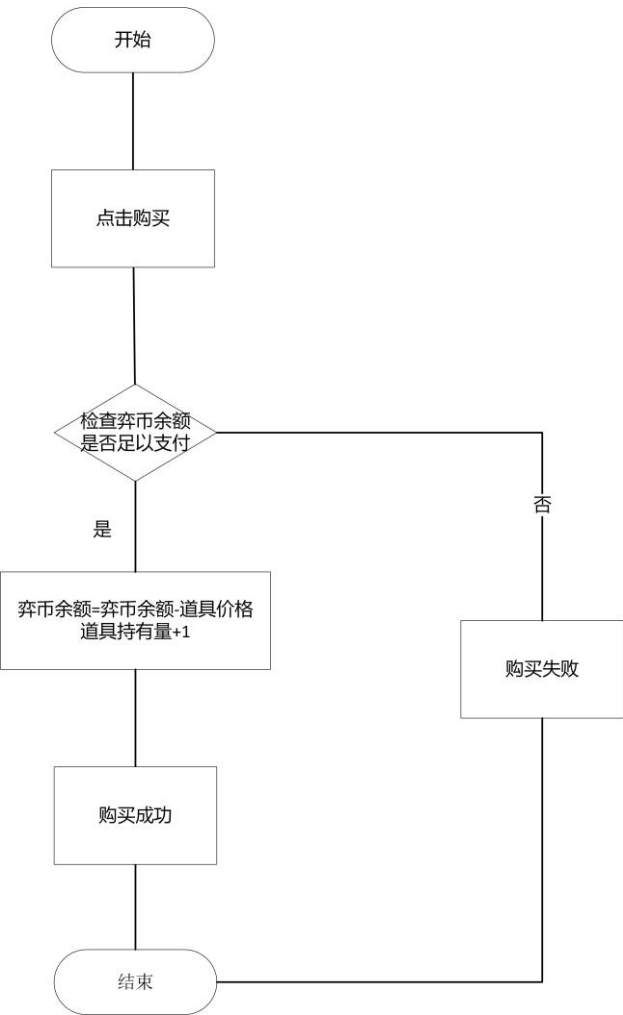


图 5-1 弈币购买道具算法设计