

Algorithm

南方小智

2020 年 11 月 21 日

目录

第一章 排序	5
第二章 贪心	6
第三章 搜索	7
3.1 DFS 深度优先搜索	7
3.2 BFS 宽度优先搜索	7
3.3 迭代深搜	7
第四章 数据结构	8
4.1 树状数组	8
4.2 字典树	8
4.3 kd 树	8
4.4 并查集	8
4.5 线段树	8
4.6 左偏树	8
4.7 点树	8
第五章 编译原理	9
5.1 逆波兰表达式	9

Table Name	Column Name	Desc/Type
DiceHistory	dice1	PostiveInt
	dice2	PostiveInt
	player	Player （哪位玩家掷的骰子）
	turn_id	Int （第几轮掷的骰子）
	(action_id)	Int (代表这个操作是该场游戏的第几个 action, 用来重播游戏过程, 用于 Debug)
	game	Game
RobberHistory	x	Int (Robber 所在位置的 x 坐标)
	y	Int (Robber 所在位置的 y 坐标)
	player	Player
	is_knight	Bool (True 代表是由 Knight 技能卡牌触发的 Robber, False 代表通过掷骰子到数字 7 触发的 Robber)
	is_latest	Bool (是否最后一个 Robber 操作, 可用于查询当前 Robber 位置, 每个游戏只有一个 true value)
	victim	Player (受害者玩家)
	(cardset_movement)	CardsetMovement (抽牌转移情况)
	turn_id	Int (第几轮发生的 Robber 事件)
	(action_id)	Int (代表这个操作是该场游戏的第几个 action, 用来重播游戏过程, 用于 Debug)
	game	Game
Cardset	lumber	Int (default=0)
	brick	Int (default=0)
	wool	Int (default=0)
	grain	Int (default=0)
	ore	Int (default=0)
	dev_knight	Int (default=0)
	dev_one_victory_point	Int (default=0)

	dev_road_building	Int (default=0)
	dev_monopoly	Int (default=0)
	dev_year_of_plenty	Int (default=0)
Player	card_set	CardSet
	order	Int (取值范围是 0 玩家数, 代表该玩家是第几个开始行动的玩家。)
	color	Chars (玩家的颜色)
	knight number	(可以根据 RobberHistory 进行计算)
	(user)	Int (用户 id, 外层的 Portal 系统负责用户的注册, 有用户名, 头像等信息。头像信息也可以放到每个 Game 里, 也就是每个游戏可以随时设置不同头像)
	game	Game
Construction	type	Chars (可选 House, Town, Road)
	owner	Player (物件所属玩家)
	x	Int (所在位置 x 坐标)
	y	Int (所在位置 y 坐标)
	z	Int (所在位置 z 坐标)
	game	Game
Tile	type	Chars (可选五种基本资源, Sea, Desert)
	number	Int (每个地块上的数字, 2 ~ 12)
	x	Int (所在位置 x 坐标)
	y	Int (所在位置 y 坐标)
	game	Game
HarborSea	type	Chars (可选五种基本资源, Any3)
	x	Int (所在海洋位置 x 坐标)
	y	Int (所在海洋位置 y 坐标)
	game	Game
	x	Int (所在位置 x 坐标)
HarborLand	y	Int (所在位置 x 坐标)

	z	Int （所在位置 x 坐标）
	sea	HarborSea
	game	Game
Bank	cardset	Cardset
	game	Game
Game	map_name	Chars （游戏用的地图模版名）
	turn_id	Int (当前是该场游戏的第几个回合, base=0, 每个回合可能有多个 action, 前 2N 个回合为 Settle 阶段, N 为玩家数)
	status	Chars (游戏当前的阶段, 包括 settle: 放房子阶段。main: 主游戏阶段。end: 游戏结束显示结果阶段。)
	number_of_player	Int (总玩家数量, 也可以从 Player 表计算)
	(action_id)	Int (当前是该场游戏的第几个 action)
	curr_player	Player （当前玩家）

表 1: 数据库设计

第一章 排序

第二章 贪心

第三章 搜索

3.1 DFS 深度优先搜索

3.2 BFS 宽度优先搜索

3.3 迭代深搜

第四章 数据结构

4.1 树状树组

4.2 字典树

4.3 kd 树

4.4 并查集

4.5 线段树

4.6 左偏树

4.7 点树

第五章 编译原理

5.1 逆波兰表达式