





# PART ONE

游戏规则简介



#### 游戏规则



把相同且相挨着**图案连起** 来消掉

消掉之后出现空隙,那不相挨着的也可以消掉。但 两个选中的方块之间连接 线的折点不超过两个(由 与X轴和Y轴的平行线组成)

# PART TWO

代码简介



#### **MORE THAN TEMPLATE**

```
settings.points != []:
 settings.points.append(btn)
# 检查是否相同、能否相连
 if can_clear(settings.points):
    # 消灭它们
    for point in settings.points:
        print("+++++++++", point.number)
        print("----, map_list)
        map_list[point.number] = 0
        point.number = 0
        point.hide()
else:
    # 不匹配,恢复图片状态
    for point in settings.points:
        point.reset()
# 这次判断完毕,清除记录的点
 settings.points.clear()
```

先点击一个元素,并记录元素。再通过点击第二个点的 坐标,来判断是否相同。



#### **MORE THAN TEMPLATE**

3

4

每种方块个数随机

先按顺序将每种图片排好放入列表 tmp\_list中

然后用random打乱列表元素的 顺序

最后依次从 tmp\_list中取一个图标方块放入m list中TE

```
# 构建成对数据
for i in range(0, settings.map_total, 2):
   # 随机生成成对的图片元素标号(1-12), 存放于 tmp_list
   e = math.ceil(random.random()*settings.element_num)
   # double append
   t list.append(e)
   t list.append(e)
# 打乱数据
for i in range(0, settings.map_total, 1):
   #将 tmp_list 中的图片元素随机排列在 m_list
   index = int(random.random()*(settings.map_total-i))
   m list.append(t list[index])
   # 删除已保存到 m_list 中的元素
   t_list.pop(index)
return m list
```



0

### 联通算法



# PART THREE

实战视频





## 自画小人









#### 实战视频





#### 不足与反思

没有画出两个元素相消时的轨迹,不够直观

如果游戏陷入死局,则无法判断

界面不够精美,无bgm等

游戏缺乏创新性

