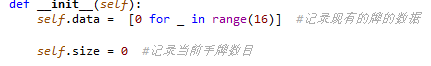
数算大作业报告

小组成员：刘潞、曹义彬、董志金、石家豪

实现类说明：

：利用列表记录当前手牌，同时方便地进行出牌计算

内置函数说明：

：初始变量包括，所拥有的手牌&手牌数目

：向现有的手牌中加入一个牌组，输入为一个列表，其中包含我们加入的手牌字符，无输出。

*：*将现有的手牌移除部分牌组，输入为一个列表，其中包含我们减少的手牌字符，无输出。

*：将目前的手牌以列表形式输出*

：为对方所出的牌组，输出针对对方的牌组我方所能进行回应的牌组的集合，例：我方现有手牌 。 特别说明：对于对方所处的三带一、三带二、三连（带牌）情况，会对回应牌组中所带的牌进行一定的计算，尽可能减少拆散的顺、对、连牌型。

实现外置函数说明：

：判断一个牌组是否能一手出完，如能一手出完，返回该牌组的类型；如不能则返回。例：

：为实现TotalHand的一个雏形函数，功能是找出可以出的三连对，三同张，以及顺子，单牌等众多牌型，并返回其集合的列表，之所以说是雏形，是因为这一函数并不能实现三连带牌的情况，但是我们可以借助这个函数实现另一个有助于我们判断所带的牌的函数NeedRound

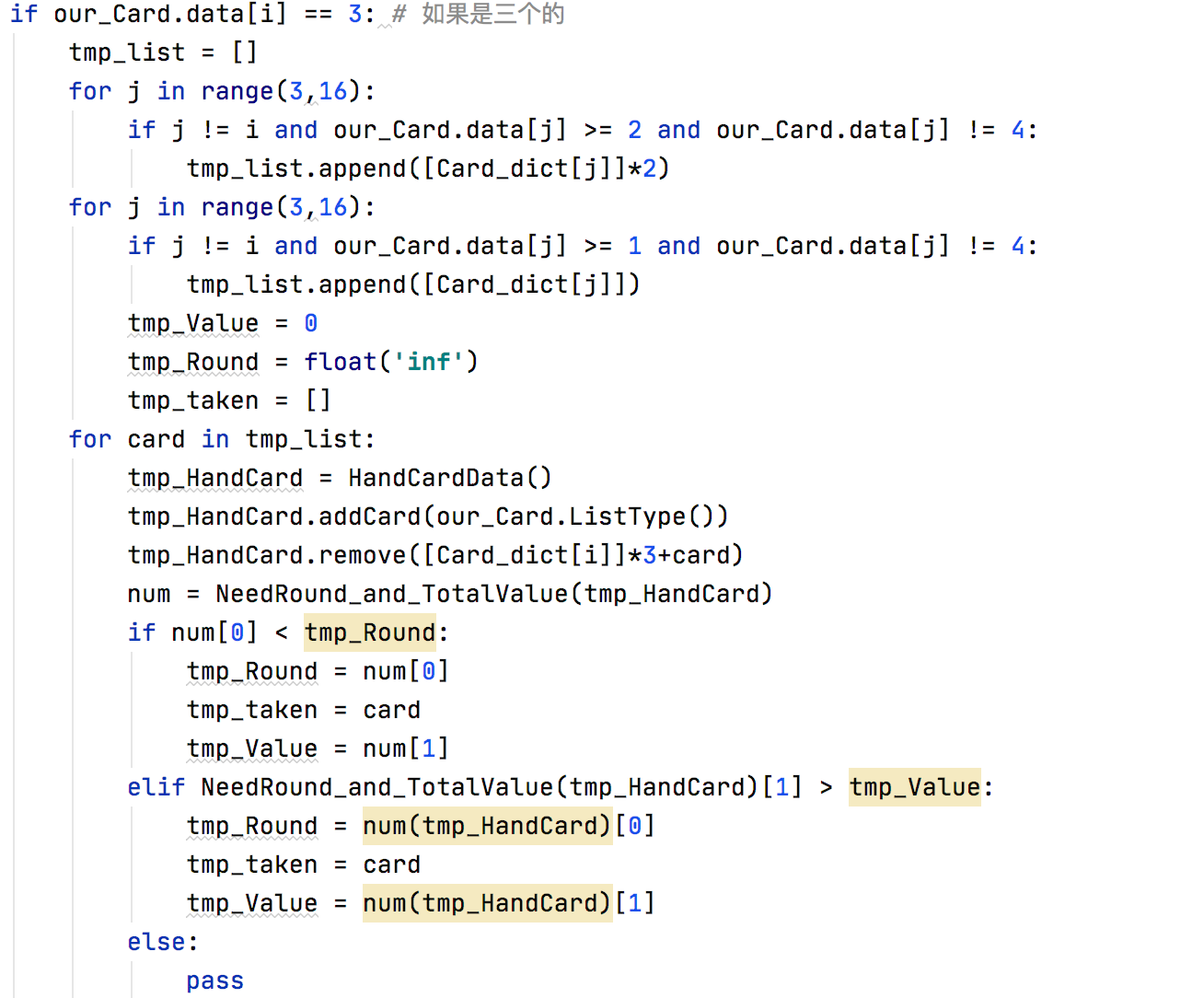
:利用每次从TotalHand\_（即TotalHand的雏形函数）中提取最长的可出牌的子串，并利用递归来判断出完手牌需要的轮数，本质上是贪心算法，利用比较容易找出的最长的牌，来判断当前牌最少需要几轮出完。这样在我们应对三带一、三带二等牌时，可以将能带的牌依次进行判断，找出剩余轮数最少的出法，核心思想其实是我们在斗地主的时候带牌尽量不要拆对子、顺子、三同，然后尽量找小的出。

屏幕快照 2020-05-26 下午4.55.30判断自身目前的手牌能够出的所有牌的组合，并且数量由多至少进行排列。其中包括的类型有顺子、三连、三带一、三带二、双连等规则中允许的所有类型。其中实现了一些比较特殊的功能：（1）三连的带牌：最简单的思路是我们用三连去带手牌中最小的牌。但是我们考虑到这种情况可能会拆散一下特殊的类型，比如顺子、双顺等等，导致牌型被打乱。所以我们在实际操作中对所能选取的可能性进行遍历，如果选取某种情况时会使得除三连外的牌组数量大减或者全部出完的手数增多则继续遍历。



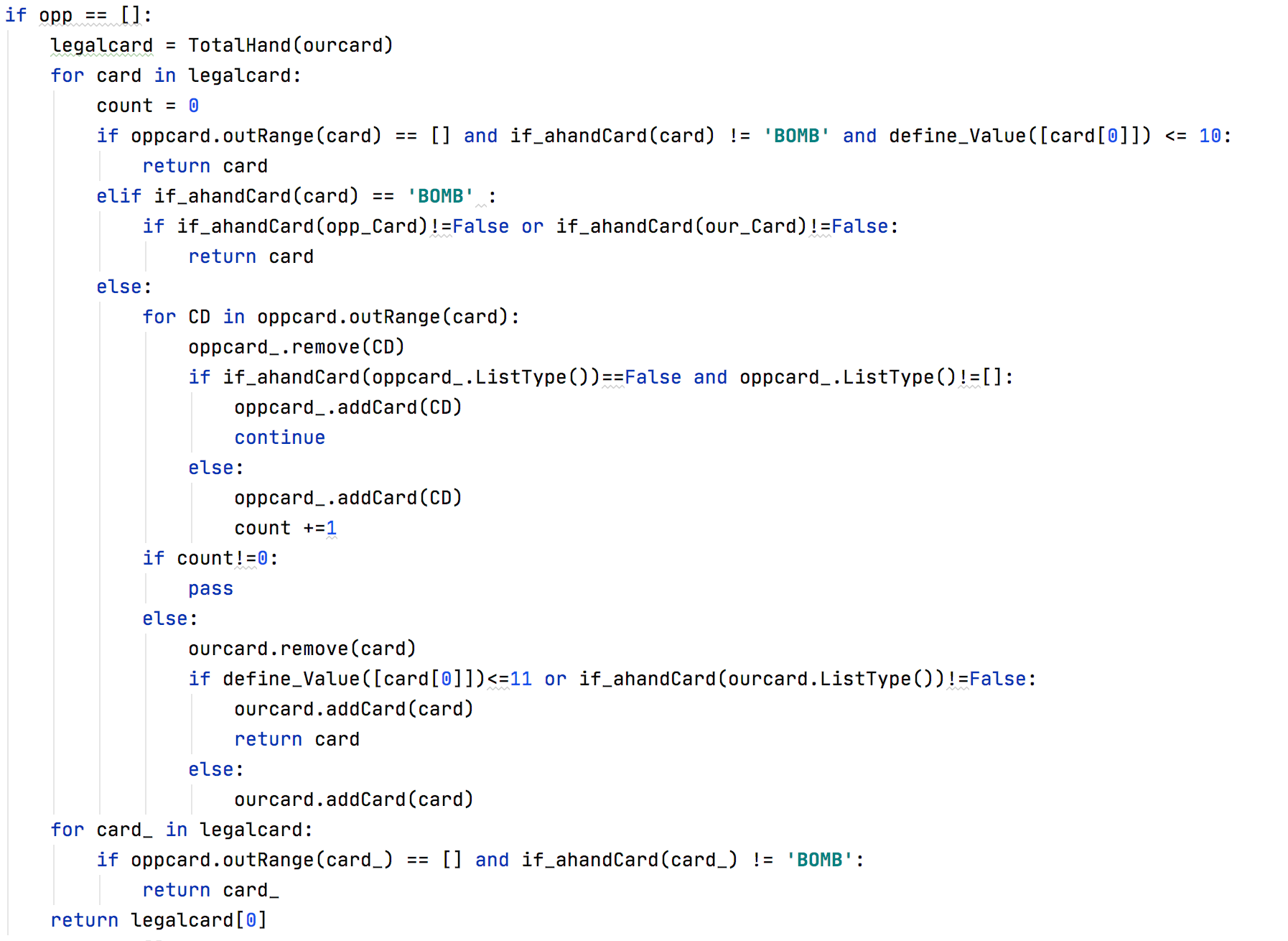
图一：三连的组合选择

（2）三带一与三带二的带牌：此种情况与上述面临着同样的问题。此时我们运用外置的needround函数来判断每一个带牌选择之后剩余的出牌手数，选择一个最小的来作为自身出牌的选择。



图二：三带一的选择

屏幕快照 2020-05-26 下午5.08.48这个函数是我们的主体出牌函数，通过输入我们的手牌，对方的手牌与我们将要应对的牌来进行输出我们的出牌。其中函数内主要分为两个函数：我方主动出牌与我方被动接牌。在我方主动出牌时：（1）对方无法接：此时我们具有相对的主动权，所以如果这类牌的数量较多，价值较小的话我们就直接出此类牌。（2）对方可以接：此时我们会让对方继续出一手牌，此时我们要进行判断对方再出一手牌之后能否变成一手牌。若会变成的话则应考虑其他选择。



图三：主动出牌

在被动出牌时，我们会在所能出的选择中选择剩余牌组手数最少，并且价值也相对合理的一种选择。也就是并不一定要出选择中最小的牌数，如果仅仅这样简单考虑的话我们将会拆散一些较为整齐的牌组，所以我们会对剩余牌数的手数也做一定的考虑。



图四：被动出牌