前面的两个函数answer\_analyze和input\_analyze，用两个数组记录answer和guess字母的出现情况。紧接着outputlater和outputformer通过比对两个数组得到基础要求中所需的输出结果。基础要求2多局游戏多次循环单局游戏，每次对所用数组进行初始化。基础要求3在检测猜测词合法性上提出新要求，我利用了一个数组和string记录之前上一次猜测词的情况，与新的猜测比较，判断是否满足新要求。1）在提高要求：筛选可能答案上，我遍历了词库，将其作为可能答案，猜测词不变，看看所得到的输出结果是否与原有一致。因为筛选可能答案在一局游戏中需要多次进行，因此还需与原有的可能答案库求出交集。另外，题目中要求筛选出所有可能答案，这在最初几次猜测中选择很多，因此我才用了十个单词换行的方式，使输出看上去更清晰一些。2）在计算信息熵时，我利用了3blue1brown视频中的算法，并建立一个vector pair, 计算出每个可选词的信息熵作为第二位并据此由大到小排序。3）在交互解决WORDLE时，让用户输入第一个猜测词的字母状态，给出不同的答案会让该猜测词出现多少种字母状态，即colorings的数量；以及让猜测词出现该字母状态有多少个可能答案，并给出下一步的最优猜测词。再让用户输入这个词的字母状态，继续上一步的操作，直到获得最终答案，或者猜测词数用光。4）遍历答案词库，每次猜测信息熵最大的词，统计所用的平均猜测数。其中，第一步所需的运算量较大。因为第一步需要计算的词语范围都是整个acceptable\_set，算出的信息熵最大始终是tares, 这与3blue1brown中说明的一致。最后遍历了全部词语的信息熵并排序，选出前十个作为全局最优的起始猜测词。