

2048

【问题描述】

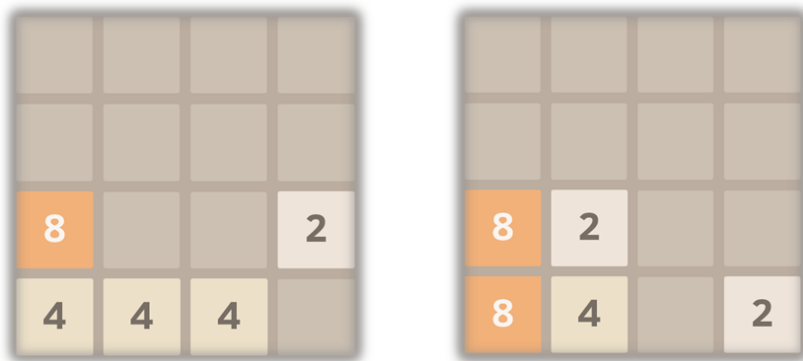
2048 是最近十分风靡的一款小游戏。

我们先介绍一下实际游戏的规则：2048 游戏由 4×4 的矩阵构成。每个位置（方格内）可以放置不多于一个瓦片(tile)，每个瓦片上标着一个值为 2 的幂的数。玩家每回合从上下左右中选择一个**合法**的方向（合法性将在稍后解释），这些瓦片将**尽可能**向该方向移动，直至边界，并且保持顺序。若相互**挤压**（即**相邻**且**相对位置平行**于所选方向）的瓦片数字相同，则立即合并为一个瓦片，消失的瓦片立即不占据位置。每回合每个瓦片**仅可合并一次**，并且合并次序以沿玩家所选方向**靠前者**为先。一个方向**合法**当且仅当该选择的方向至少移动或合并一个瓦片，因此该操作后，矩阵上至少存在一个空白位置。开始游戏前或回合间，游戏程序会在空白处随机生成一个标着 2 或 4 的瓦片。当场上合成 2048 时游戏结束，玩家胜利；若玩家不能做出合法的移动，游戏结束，玩家失败。

现在，我们想让 AI 尽量获得更高的分数，请注意，**本题中规则与实际游戏有所不同**。游戏中玩家不知道随机生成瓦片的位置和大小，在本题中，我们**给定随机算法**，并且每次只生成标着 2 的瓦片，希望依此获得相当高的分数。

游戏规则样例：

出现左图所示的状态后，玩家选择向左移动，下方的三个 4 将会变为一个 8 和一个 4，由于左边两个 4 更靠近移动方向（左），故应合成最左侧一个 8 和第二左侧一个 4（见右图），由图可以判断出，游戏程序在回合间随机生成了一个 2 在右下角：



随机算法如下：

给定随机种子 $seed$ ，和常量 $MUL = 8221$ ，一个有符号 32 位的随机整数数列 seq 按如下规则生成：

$$seq_0 = seed$$

$$seq_i = (seq_{i-1} * MUL) + (seq_{i-1} \gg 16) \quad (i \geq 1)$$

其中 \gg 符号表示带符号右移。当加减法结果 $\geq 2^{31}$ （或 $< -2^{31}$ ）时，自动变为模 2^{32} 意义下相等大小的负数（或正数）【Pascal 选手可以使用 $\{SR-\}$ 选项】。

2048 游戏所需的随机数为 0~15 的数字，分别代表从左到右、从上到下的依次编号的瓦片（见下图）。每次需要随机生成瓦片时，我们在随机数数列中逐个

取数并对 16 取模，直至取到一个对应空当（而非已被瓦片占据的位置）的数为
止。下次取随机数时应从上次选取数字的下一个开始。第一个随机数为 seq_1 。

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

典型的使用 C 和 Pascal 生成上述随机数序列的代码如下：

```
int MUL = 8221;
int seq;

void initSeed(int seed)
{
    seq = seed;
}

int getRand()
{
    return (seq = (seq * MUL) + (seq >> 16)) & 15;
}
```

```
Const
    MUL = 8221;
    ArraySize = 1048576;
Var
    Seq: Array[0..ArraySize] of Longint;
Procedure Init(Const Seed: Longint);
Var
    i: Longint;
Begin
    Seq[0] := Seed;
    For i := 1 to ArraySize do
        Seq[i] := (Seq[i - 1] * MUL) + (Seq[i - 1] Shr 16) + Longint(Seq[i - 1] < 0) * $FFFF0000;
    End;
Function GetRand(Const i: Longint): Longint;
Begin
    GetRand := Seq[i] and 15;
End;
```

输出文件中请按顺序包含一组合法的操作，尽可能获得标记数字更大的瓦片。
评分时将根据执行完毕后的**最终状态**评定。

附加文件 `simulate(.exe)` 可以检查您的输入文件并进行模拟，输出其最终状态
到控制台。

使用方法：在 terminal 或 `cmd.exe` 中调用 `simulate [filename: 2048*.out]`。

【输入格式】

本题没有输入文件，但我们在这里指定一组随机种子，第 i 题使用数值为 i 的
随机种子 ($i = 1, 2, \dots, 20$)，对应输出文件分别为 `20481.out~204820.out`。

【输出格式】

在输出文件 `20481.out~204820.out` 中，第一行输出该题目对应的随机种子（注
意文件名和种子顺序不要交换）。第二行按顺序列出操作（上下左右分别用大写

字母 UDLR 表示)。

【样例输入】

(无)

【样例输出】

```
1
LLLLLDLLLRDULRRDRUUDRLUURLURLLUDLLURLRLLULLLLL
```

【样例说明】

以上的样例是 20481.out 的一个合法输出。它可能并不能得到这个测试点的满分。

【评分标准】

每个测试点单独评分，每个测试点权重相等。如果你的输出不合法或不满足要求，得分为 0，否则得该测试点满分。要求如下。

对于测试点 1-2：你的**最终状态**出现标记不小于 4096 的瓦片。

对于测试点 3-5：你的**最终状态**出现标记不小于 8192 的瓦片。

对于测试点 6-10：你的**最终状态**出现标记不小于 16384 的瓦片。

对于测试点 11-20：你的**最终状态**出现标记不小于 32768 的瓦片。