

火柴棒游戏设计报告

3180106071 刘轩铭 软件工程

设计目标和完成情况

设计内容

火柴棒游戏

- 用户从命令行输入最大数字的位数（如1位数、2位数、3位数）；
- 用户从命令行输入提示数（2或3），表示等号左式数字的个数；
- 用户从命令行输入题目类型编号（移动、移除、添加），以及火柴棒根数；
- 系统随机自动生成火柴棒游戏，并展示（直接用数字展示）；
- 用户输入答案，系统验证是否正确；若正确，则提示正确；若错误，则让用户继续输入；
- 若用户直接回车，则显示正确答案。

任务实现与亮点

- ☑ 游戏的完整实现
- ☑ 用真实的火柴形式显示算式和结果
- ☑ 拥有效率较高的随机算法生成题目
- ☑ 可以实现操作10根火柴棒这种数量的题目生成
- ☑ 对多种情况都进行了测试，结果均良好（见测试部分）
- ☑ 较高的代码质量、较多的注释和较好的封装效果

设计思路

类的设计

Main

- 主循环，游戏主体部分
- 内容如下：

```
// ResultType is a enum class that defines the status of answer correctness.
public class Main {
    public static void main(String[] args) throws Exception {
        while(true) {
            UserInterface disp = new UserInterface();
            disp.displayInputConditionPrompt();
            FunctionInterface bar = new
FunctionInterface(disp.getUserCondition());
            bar.innerGenerateProblem();
```

```

        while(true) {
            disp.userInput();
            ResultType res = bar.ansCheck(disp.getUserInputAns());
            if(res == ResultType.RIGHT)
                disp.rightAnsPrompt();
            else if(res == ResultType.ACQUIRE_CORRECT_ANS)
                disp.printCorrectAns(bar.getEquationOfAns());
            else disp.wrongAnsPrompt();
            if(res == ResultType.ACQUIRE_CORRECT_ANS || res ==
ResultType.RIGHT) break;
        }
    }
}
}
}

```

FunctionInterface

- 公有类，负责进行题目的生成，检验等功能性管理
- 拥有下面一些成员变量：

```

private int maxBitsNumOnLeft, amountOfNumOnLeft, typeOfProblem,
numOfMatchsticks;
private char[] equationOfAns = new char[EQUATION_LEN];
private char[] equationOfProblem = new char[EQUATION_LEN];

```

- 拥有下面一些主要方法：

```

public void innerGenerateProblem();

public ResultType ansCheck(String in);

public void innerGenerateProblem();

public ResultType ansCheck(String in);

private void randGernerateCorrectEquation();

private void randGernerateWrongEquation();

```

UserInterface

- 公有类，负责和用户进行交互，读取和显示信息
- 拥有下面一些成员变量：

```

private int[] userInputCondition = new int[INPUT_NUM];
private String userInputAns = "";

```

- 拥有下面一些主要方法：

```

// constructor.

```

```

    UserInterface();

    public void displayInputConditionPrompt();

    public int[] getUserCondition();

    public String getUserInputAns();

    public void rightAnsPrompt();

    public void printCorrectAns(char[] correctAnsEquation);

    public void wrongAnsPrompt();

    public void userInput();

```

整体设计思路

要完成本游戏，至少需要以下的几个部分：

- 首先需要有一个大的循环，内部不断地进行游戏
- 在每个循环内，需要打印输入条件提示信息，生成题目，打印题目，用户输入答案和程序验证答案这几个步骤
- 为此，我设计了两个类，分别是 **FunctionInterface** 和 **UserInterface**，负责题目内部功能的实现和对用户展示信息的实现
- 在上述的步骤中，打印输入提示信息是比较好完成的，只需要在 **UserInterface** 类中设计若干方法就可以了。我设计的若干方法如下：

```

    UserInterface();
    public void displayInputConditionPrompt();
    private boolean isLegalInput(int[] userInput);

```

在这里实现了读取四个整数，并判断合法性，然后将整数存入 **Userinterface** 的成员变量中。

- 其他几个步骤的实现将在下面详细说明

生成原题目

- 在信息读取过程中，题目要求的四个参数被保存到 **FunctionInterface** 类的四个成员变量 **maxBitsNumOnLeft**, **amountOfNumOnLeft**, **typeOfProblem**, **numOfMatchsticks** 中
- 随后，**FunctionInterface** 调用 **randGernerateCorrectEquation** 和 **randGernerateWrongEquation** 两个方法来生成题目
- 首先对 **amountOfNumOnLeft** 进行分类，分别对左边有两个操作数和三个操作数的情况进行等式的生成
- 在每种情况内，使用随机数函数，根据 **maxBitsNumOnLeft** 是1，2或是3来确定生成数字的范围是一个一位数，两位数，或是三位数。这样可以生成2/3个操作数和1/2个操作符，然后根据左边的算式，计算出答案作为右式
- 接下来调用 **private void setOneNumOrSignIntoEquation(int num, int begin, boolean isSign);** 方法将生成的操作数和操作符存入 **equationOfAns** 中。我们最多有三个三位数的操作数，一个三

位数的结果和三个操作符（包括等于）的情况。故一般形式为 $xxx+xxx+xxx=xxx$ ，所以我用一个15位的char[]来存储这个等式。如果某一位不存在，我们用 '*'来代替

- 这时候我们已经有了一个正确的等式，接下来通过随机算法将其改为不正确的式子

生成题目

根据题目要求，有添加，删除和移动三种情况，我们分别进行说明。在

randGenerateWrongEquation 方法中，我们也根据 typeOfProblem 变量来进行情况的划分。

添加

- 由七段数码管表示数字得到启发，我们给出如下的一个哈希表，来调代表每个数字对应的火柴数量和位置：

```
/**
 * In seven-segment display,
 * A number is displayed as below:
 *      a
 *      *****
 *      *      *
 * f *      * b
 *      * g *
 *      *****
 *      *      *
 * e *      * c
 *      * d *
 *      *****
 *
 * which can be represented using binary strings as:
 * 1111111
 * with an one-to-one correspondence to
 * abcdefg
 *
 * to make the plus sign '+', the minus sign '-' and the equals sign
 '=' able to be represented,
 * we add 5 segments on the left, then it can be represented as:
 *
 *      a      f
 *      *****
 *      *      *      *
 * b *      k*      * g
 *      * e *      *
 *      *****
 *      *      *      *
 * c *      j*      * h
 *      *      *      *
 *      *****
 *      d      i
 */
private final HashMap<Character, Integer> stickNumber = new
HashMap<Character, Integer>(){
    {
        put('0', 0b000001111110);
        put('1', 0b000000110000);
        put('2', 0b000001101101);
```

```

        put('3', 0b000001111001);
        put('4', 0b000000110011);
        put('5', 0b000001011011);
        put('6', 0b000001011111);
        put('7', 0b000001110000);
        put('8', 0b000001111111);
        put('9', 0b000001111011);
        put('+', 0b000010000111);
        put('=', 0b100101001000);
        put('-', 0b000010000001);
        put('*', 0b000000000000);
    }
};

```

- 另外还有一个二维数组，储存不同的数字和符号之间，如果发生转化，需要改变的火柴数量：

```

// The number difference of match sticks during making a change.
private final int [][] diffDuringAddOrRemove = {
    // diffDuringAddOrRemove[i][j] means the number of sticks needed
    // during changes of math number from i to j
    // 0 1 2 3 4 5 6 7 8 9 + - =
    { 0, -4, 0, 0, 0, 0, 0, -3, 1, 0, 0, 0, 0, 0},
    { 4, 0, 0, 3, 2, 0, 0, 1, 5, 4, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0},
    { 0, -3, 0, 0, 0, 0, 0, 0, -2, 2, 1, 0, 0, 0},
    { 0, -2, 0, 0, 0, 0, 0, 0, 3, 2, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 0, 1, 0, 2, 1, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, -1, 0, 0, 1, 0, 0, 0, 0, 0},
    { 3, -1, 0, 2, 0, 0, 0, 0, 4, 3, 0, 0, 0, 0},
    {-1, -5, -2, -2, -3, -2, -1, -4, 0, -1, 0, 0, 0, 0},
    { 0, -4, 0, -1, -2, -1, 0, -3, 1, 0, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0},
    { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0},
    { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
};

```

- 于是我们可以用以下的算法来生成题目：

```

01. // Add algorithm:
02. randomAlgorithm() {
03.     currentTotalSticks := 0;
04.     While currentTotalSticks < NumOfSticks:
05.         randomly generate a position index which 0 <= i <= EQUATION_LEN; // as a index of the character to be changed.
06.         j := s[i];
07.         choose a minimum number of diffDuringAddOrRemove from row j, and k := the column;
08.         change the character from j to k;
09.         currentTotalSticks += | diffDuringAddOrRemove[j][k] |
10.     }

```

主要思想是，每次从等式的15位中随机出一个位置，然后，得到对应位置的字符代表的数字或符号。在二维数组中找到那一行（这一行是该数字或符号转化为其他数字符号所需要变化的火柴数量）。从这一行里找到一个最小值（绝对值最大），然后将该数字/字符转化为对应列所代表的数字/字符。这样循环，直到题目给出的 numOfMatchSticks 用完。为了防止在同一位置多次变化，我们将使用过的位置纳入集合中，在每次变化之前判断，该位置是否已经在集合中，如果已经在，那么我们重新找一个位置。

删除

- 该情形和添加其实是类似的，我们只需要将上述算法中的找寻最小值，换成找寻最大值就可以了。

```

01. Remove Algorithm:
02. removeAlgorithm() {
03.     currentTotalSticks := 0;
04.     While currentTotalSticks < NumOfSticks:
05.         randomly generate a position index which 0 <= i <= EQUATION_LEN; // as a index of the character to be changed.
06.         j := k;
07.         choose a maximum number of diffDuringAddOrRemove from row j, and k := the column;
08.         change the character from j to k;
09.         currentTotalSticks += diffDuringAddOrRemove[j][k] ;
10. }

```

移动

- 我们将移动看成是先添加再删除，这样只需要连续调用上述过程就可以完成需要的效果了。
- 为了避免多次移动同一火柴，我们将移动过的位置同样纳入集合，来避免这种情况
- 在这三种情况的最后，我们都把生成的式子保存到 equationOfProblem 字符数组中，它和前面所说的 equationOfAns 数组是类似的。

展示题目

- 我们不但有通过打印字符数组来显示题目的方法，还可以用火柴棒表示数字，来显示题目
- 用火柴棒表示数字的方法是，建立一个二维数组，对等式字符数组进行遍历，将每一个字符填充到大型二维数组中。使用上面提到的 stickNumber 哈希表，找到每一个字符对应的二进制串，通过判断每一位上是0还是1，来决定对应位置是否填充。
- 在注释中我也详细写了对应的实现思想：

```

/**                                                                                                     ▲ 42 ▲ 4 ✖
 * the equation is: * * * + * * * + * * * = * * *,
 * where * is a number/sign represented as:
 *
 *      a      f
 *      *****
 *      *      *
 * b *      k*      * g
 * * * e * l *
 *      *****
 *      *      *
 * c *      g*      * h
 *      *      *
 *      *****
 *      d      i
 *
 * If we have the sticks represented by "****" for each stick,
 * with 7 numbers(each contains 3 bits) and signs, 15 in total, each occupies 9 rows and 9 columns,
 * plus 14 spaces, each occupies 3 columns,
 * we obtains a multi-dimension array with ((STICK_LEN*2+3)*7+SPACE_LEN*6)=81 columns and (STICK_LEN*2+3) rows.
 */

```

用户输入答案

- 接下来循环输出提示信息，帮助用户输入答案
- 用户将答案输入，然后 FunctionInterface 再通过调用输入的字符数组，来进行答案的检查
- 如果答案正确，输出相应信息提示，开始下一轮游戏；如果答案错误，提示用户错误，然后重新输入；如果用户输入一个回车，则会将正确答案直接显示

答案检验

- 用户的答案字符数组传入 FunctionInterface 后，将在 ansCheck 方法内进行检验
- 检验首先对答案进行格式化：这里我们首先用replace方法将多余的空格去除，然后进行字符串匹配，找到答案中必须包含的 '=' 符号，将答案左右分开，然后根据检验左右是否符合格式。同时将进行答案格式化，将其变为同样 EQU_LEN 长度的数组

- 之后将格式化的输入答案进行正确性检验，这里有几种情况：
 - 如果格式化答案和 equationOfAns 完全一致，则返回答案正确
 - 如果不一致，则检验等式是否成立，如果不成立，返回答案错误
 - 如果成立，根据题目的 type 和操作火柴棒的数量，对答案每一位进行火柴棒数量的计算，和原来位置进行比较，如果整个等式和题目相比火柴被操作的类型和数量都是一致的，那么也可以返回正确，否则返回错误

运行结果与分析

一般情况展示

```
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 2 1 2

      **      **      **      **      **      **      **
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
      **      **      **      **      **      **      **
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
      **      **      **      **      **      **      **

Please input your answer or input a single ENTER to display the correct answer:
10-123= 23
Wrong Answer!
Please input again.
Please input your answer or input a single ENTER to display the correct answer:
```

```
The correct answer is:
137+537=674
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
```

- 以上是一次正常的游戏，用户尝试失败后，显示正确答案并进行下一轮游戏

```
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 2 1 3

      **      **      **      **      **      **      **
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
      **      **      **      **      **      **      **
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
      **      **      **      **      **      **      **

Please input your answer or input a single ENTER to display the correct answer:
*314+33 - 4=1ok0
Wrong Answer!
Please input again.
Please input your answer or input a single ENTER to display the correct answer:

The correct answer is:
100+394=502
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
```

- 故意输入格式不正确的答案，也会显示输入错误，证明格式判断是正确的

```
new game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
1 1 3 1
Illegal input, please input again.
```

- 非法的输入，进行提示并重新输入

```
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 3 1
**      **      **      **      **      **      **      **      **      **      **      **
*  *  *      *      *  *  *      *      *      *      *      *      *      *
*  *  *      *      *  *  *      *      *      *      *      *      *      *
**      **      **      **      **      **      **      **      **      **      **      **
*      *  *      *      *  *  *      *      *      *      *      *      *      *
*      *  *      *      *  *  *      *      *      *      *      *      *      *
**      **      **      **      **      **      **      **      **      **      **      **
Please input your answer or input a single ENTER to display the correct answer:
207+862-332=737
Correct Answer!
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
```

- 这是一次正确的输入，显示相应信息并重新开始游戏

移动情况展示

```
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 1 3
**      **      **      **      **      **      **      **      **      **      **      **
*  *  *      *      *  *  *      *      *      *      *      *      *      *
*  *  *      *      *  *  *      *      *      *      *      *      *      *
**      **      **      **      **      **      **      **      **      **      **      **
*      *  *      *      *  *  *      *      *      *      *      *      *      *
*      *  *      *      *  *  *      *      *      *      *      *      *      *
**      **      **      **      **      **      **      **      **      **      **      **
Please input your answer or input a single ENTER to display the correct answer:

The correct answer is:
794-245+297=846
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
```

- 这是一次普通的移动三根火柴棒的游戏

删除情况展示


```
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 2 4
  **      **      **      **      **      **      **      **      **
*  *      *      *      *      *      *      *      *      *
*  *      *      *      *      *      *      *      *      *
  **      **      ** **      **      **      **      **      **
*  *      *      *      *      *      *      *      *      *
*  *      *      *      *      *      *      *      *      *
  **      **      **      **      **      **      **      **      **
Please input your answer or input a single ENTER to display the correct answer:

The correct answer is:
972+24-958=38
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
```

- 这是一次普通的删除三根火柴棒的游戏

添加情况展示

```
The correct answer is:
972+24-958=38
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 3 5
  **      **      **      **      **      **      **      **      **
*  *      *  *      *      *      *      *      *      *      *
*  *      *  *      *      *      *      *      *      *      *
  **      **      ** **      **      **      **      **      **
*  *      *      *      *      *      *      *      *      *
*  *      *      *      *      *      *      *      *      *
  **      **      **      **      **      **      **      **      **
Please input your answer or input a single ENTER to display the correct answer:
```

- 这是一次普通的添加三根火柴棒的游戏

大数据情况

```

This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 3 10

```

```

Please input your answer or input a single ENTER to display the correct answer:

The correct answer is:
461+273-467=267
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:

```

- 一次添加十根，也能跑出结果

```

This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:
3 3 1 10

```

```

Please input your answer or input a single ENTER to display the correct answer:

The correct answer is:
313+865-373=805
This is a matches game.
New game:
Please input the bit number of the max number: (1~3)
Please input the number of matchstick numbers' amount on the left: (2 or 3)
Please input the type of the problem: (1 for moving, 2 for removing and 3 for adding)
Please input the number of matchsticks to operate:

```

- 一次移动十根，也能跑出结果

心得与体会

- 此次实验是第一个 Java 工程项目，代码量在 800 行左右，对 Java 的基本语法进行了掌握。我有一些体会：
- 第一，我不止学习和熟练了 Java 的基本语法，还掌握了一些进阶知识，如 HashMap 等的用法，自我学习还是一个很重要的技能。
- 第二，要提前想好代码的结构。这一次我重点对代码结构进行了封装和调整，尽量避免了代码的重复等，也什么有成就感。
- 第三，算法思想很重要。我一开始也想过通过暴力枚举的方法来生成题目，但是这样的效率可能更低，所以选用了七段数码管的思想，对每个位进行随机。通过此次实验，也收获颇多，以后要重点学习 Java 的进阶思想了。