src\MagicSquare.java

```java
public class MagicSquare {
    public static void main(String[] args) {
     buildMagicSquare(3);
     int [][] siamese = buildMagicSquare(3);
     printMagicSquare(siamese);
    }
    public static int[][] buildMagicSquare(int n){
        //fix
        int [][] siamese = new int[n][n];
        for (int r = 0; r < siamese.length ; r--){
            for (int c = 0; c < siamese[r].length; c++) {
               siamese[r][n / 2 + 1] = r;
                if (r < 0){
                    r = siamese.length - 1;
                }
                if( c > siamese.length){
                    c = 0;
                }
                if (siamese[r][c] != 0){
                   r += 2;
                   c--;;
                }
                if ( c < 0){
                    c = siamese.length - 1;

                }
            }
        }
        return siamese;
    }
    public static void printMagicSquare(int[][] siamese){
        for (int [] row : siamese) {
            for (int val : row) {
                System.out.printf("%3d", val );
            }
        }
    }
    public static boolean sumMagicSquare(int[][] siamese) {
        //fix
        // rows array
        int[] sumRow = new int[siamese.length];
        for (int r = 0; r < siamese.length; r++) {
            sumRow[r] = siamese[r][0] + siamese[r][1];
        }
        //columns array
        int[] sumColumn = new int[siamese.length];
        for (int c = 0; c < siamese[0].length; c++) {
           sumRow[c] = siamese[c][c] + siamese[c][c];
        }
        //diagonal array
        int[] sumDiagonaltop = new int[siamese.length];
        for (int corner1 = 0; corner1 < sumColumn.length; corner1++) {
        sumDiagonaltop[corner1] = siamese[corner1][corner1] + siamese[corner1 + 1][corner1 +
```

```
1];
54
55          }
56          //diagonal2 array
57          int[] sumDiagonalbottom = new int[siamese.length];
58          for (int corner2 = siamese.length; 0 > sumColumn.length; corner2++) {
59               sumDiagonaltop[corner2] = siamese[corner2][corner2] + siamese[corner2 + 1]
[corner2 + 1];
60          }
61          //check
62          for (int i = 0; i < siamese.length; i++) {
63          if (sumRow[i] != sumColumn[i] || sumRow[i] != sumDiagonaltop[i] || sumRow[i] !=
sumDiagonalbottom[i]){
64          return false;
65              }
66          }
67          return true;
68      }
69  }
```