



链滴

PHP 实现输入一个整数 n , 输出一个 $n \times n$ 蛇形矩阵

作者: [barryzpc](#)

原文链接: <https://ld246.com/article/1587391378382>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



前段时间群里发了一个有意思题，自己试着简单粗暴的实现一下。

题目：

实现输入一个整数 n ，输出一个 $n \times n$ 蛇形阵。例如：

```
//输入 3,输出:  
1 2 3  
8 9 4  
7 6 5
```

```
//输入 4, 输出:  
1 2 3 4  
12 13 14 5  
11 16 15 6  
10 9 8 7
```

思路：

1、观察可以发现，矩阵的值的规律是由外向内画圈依次递增的卷入（排列），是这个题的突破口。可以按照这个依次递增的值的位置来依次赋值

2、而且转的圈数为 n 除以2(也就是行数，或者是列数)，如果 n 为奇数，则向取整，即+1

代码实现：

创建php脚本文件snake_matrix.php

```
<?php

/**
 * 输入一个整数n，要求输出一个n×n蛇形阵（n<10），比如输入整数4，则输出如下蛇形阵：
 * 1  2  3  4
 * 12 13 14 5
 * 11 16 15 6
 * 10 9  8  7
 */

//收到的参数
$params = getopt("n:");
$input_num = $params['n'];
$value = 1; //矩阵的第一个值
$matrix_arr = [];

/**
 * 可以发现，矩阵的值的规律是依次递增的；这也是这个算法的突破口。
 * 而且转的圈数为 n 除以2(也就是行数，或者是列数)，如果 n 为奇数，则向上取整，即+1
 */

//圈数
$circle_num = ceil($input_num / 2);
for ($i = 0; $i < $circle_num; $i++) { //一圈一圈的来

    //用于每一圈的第一行赋值
    for ($j = $i; $j < $input_num - $i; $j++) {
        $matrix_arr[$i][$j] = $value++;
    }

    //用于每圈的最后一列赋值
    for ($j = $i + 1; $j < $input_num - $i - 1; $j++) {
        $matrix_arr[$j][$input_num - $i - 1] = $value++;
    }

    //用于每圈的最后一行赋值
    for ($j = $input_num - $i - 1; $j > $i; $j--) {
        $matrix_arr[$input_num - $i - 1][$j] = $value++;
    }

    //用于每圈的第一列赋值
    for ($j = $input_num - $i - 1; $j > $i; $j--) {
        $matrix_arr[$j][$i] = $value++;
    }

}

//按照索引依次打印出来
foreach ($matrix_arr as $key => $row_val) {
    ksort($row_val);
```

```
$str = "";
foreach ($row_val as $value) {
    echo $value;
    echo " ";
}
echo PHP_EOL;
}
```

运行php脚本

```
php snake_matrix.php -n 4
```

输出

```
1 2 3 4
12 13 14 5
11 16 15 6
10 9 8 7
```

ps: 如果大家还有别的实现方式, 可以在评论区贴出来, 大家一起学习。语言不限