

8: 方块左右移动

移动，就是左右移动，y轴不变，所以在移动的时候，只需要根据左移还是右移，去改变 $+ = 2$ 或者 $- = 2$ ，因为方形占2个字符。

```
/// <summary>
/// 按 <- ->左右移
/// </summary>
/// <param name="type"></param>
2 个引用
public void MoveLR(E_ChangeType type)//因为这个枚举刚好也是left, 和right
{
    //移动之前，先清除，否者上个画面的还在
    ClearDraw();
    //左移就是(-2, 0) 右移就是(2, 0) 这里只是相对与原点的坐标
    Position nowPos = new Position(type == E_ChangeType.Left ? -2 : 2, 0);// 1)
    //所以这里需要加上原点的坐标
    for (int i = 0; i < blocks.Count; i++)
    {
        blocks[i].pos += nowPos;//当前形态的方块整体都要移动 pos的距离
    }
    //移动之后，就需要绘制
    Draw();
}
```

```
2 个引用
public bool CanMove(E_ChangeType type, Map map)//需要地图类配合
{
    //根据传入的类型，决定左移还是右移
    //左移就是(-2, 0) 右移就是(2, 0) 这里只是相对与原点的坐标
    Position nowPos = new Position(type == E_ChangeType.Left ? -2 : 2, 0);// 1)

    Position tmpPos;
    //所以这里需要加上原点的坐标
    for (int i = 0; i < blocks.Count; i++)
    {
        tmpPos = blocks[i].pos + nowPos;//当前形态的方块整体都要移动 pos的距离

        //1、检测围墙
        if (tmpPos.x < 2 || tmpPos.x >= Game.w - 2) //y轴不会变
        {
            return false;
        }

        //2、检测动态墙体
        for (int i = 0; i < blocks.Count; i++)
        {
            tmpPos = blocks[i].pos + nowPos;
            for (int j = 0; j < map.dynamicWalls.Count; j++)
            {
                if (tmpPos == map.dynamicWalls[j].pos)
                {
                    return false;
                }
            }
        }
    }

    return true;
}
```

注意这2个地方的赋值，是+之后赋值给tmpPos临时位置节点，而不是+=，我之前因为赋值写成：
`tmpPos = blocks[i].pos += nowPos` 导致花费了大量的时间去查问题，最后通过调试发现问题。所以一定要细心