**Практика «Лабиринт диагональ»**

В том же проекте выведите робота из лабиринта вида «диагональ».

Запустите проект и самостоятельно изучите этот тип лабиринтов.

Дополнительные ограничения:

1. Запрещено использовать более одного цикла в одном методе.
2. Запрещено иметь методы длиннее 12 строк кода.
3. Запрещено использовать ключевое слово catch
4. Разрешено создавать вспомогательные методы, но только понятными именами, в том числе именами аргументов.

// Вставьте сюда финальное содержимое файла DiagonalMazeTask.cs

namespace Mazes

{

public static class DiagonalMazeTask

{

public static void MoveOut(Robot robot, int width, int height)

{

int wh, hw;

wh = (width - 2) / (height - 2);

hw = (height - 2) / (width - 2);

while (robot.Finished == false)

{

if (width > height)

{ Move(robot, wh, Direction.Right, Direction.Down); }

else

{ Move(robot, hw, Direction.Down, Direction.Right); }

}

}

public static void Move(Robot robot, int aspectRatio, Direction direction, Direction directionTwo)

{

if (!robot.Finished)

for (var i = 0; i < aspectRatio; i++)

robot.MoveTo(direction);

if (!robot.Finished)

robot.MoveTo(directionTwo);

}

}

}