# Game

public Minotaur minotaur

public Theseus theseus

public Tile[,] Map

public void CreateMap()

public void MoveMinotaur(Point theseusPos)

public bool MoveInDirection(bool isX, int theseusPos, int minotaurPos)

public bool TryToMoveMinotaur(Point direction)

public bool IsDirectionWall(Point direction)

# Thing

public Point Position

# Tile

public Walls FourWalls { get; set; }

# Walls

Up = 1,

Down = 2,

Left = 4,

Right = 8,

None = 0

# Minotaur

# Theseus

# Game

This method takes the responsibility of moving Minotaur. It takes the new Theseus’s position(X, Y), checks if Minotaur should move in X or Y direction, checks for walls then set Minotaur’s new position (Moves).

**public void MoveMinotaur(Point theseusPos)**

for Minotaur’s X coordination move

Boolean variable hasMovedX(or whatever) = MoveInDirection(true, Theseus’s X

position, Minotaur’ Y position)

for Minotaur’s Y coordination move

if (hasMovedX is false) then

MoveInDirection(false, Theseus’s Y position, Minotaur’ Y position)

This method return True or False if Minotaur can move to the specified direction. If Minotaur should move, then pass the direction parameter to next method TryToMoveMinotaur to further check if the target position is accessible(no wall to prevent).

**public bool MoveInDirection(bool isX, int theseusPos, int minotaurPos)**

if (Theseus’s position is equal to Minotaur’s position) then

return false;

if (Theseus’s position is less than Minotaur’s position) then

return TryToMoveMinotaur(Left or Up)

return TryToMoveMinotaur(Right or Down);

Calls the method IsDirectionWall to check if Minotaur can move to the specified direction. If it can, set a new Minotaur’s position.

**public bool TryToMoveMinotaur(Point direction)**

if (IsDirectionWall(direction))

return false;

*else*

set Minotaur’s new X and Y position to the target tile’s X and Y position

return true;

**public bool IsDirectionWall(Point direction)**

Validate if the target Tile actually exists.(Not outside the map)

If it is, then return true (there are boundry walls)

if Minotaur is moving Left

if target tile has Right wall

return true;

else if Minotaur is moving Right

if target tile has Left wall

return true;

if Minotaur is moving Up

if target tile has Down wall

return true;

else if Minotaur is moving Down

if target tile has Up wall

return true;

return false;