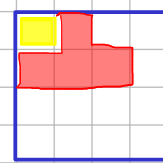
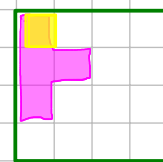


Pieces  
General.  
x/y-coords:

4x4 Piece Map (default map)



Normal Position



Rotation Piece Map  
Rotated Piece

3 Rotation maps  
+ 1 default map

1-Piece:

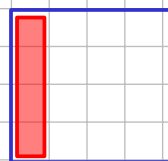
1-Piece {

x:  
y:  
rotation:  
blocks[]:

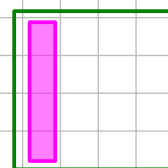
}

Impl:

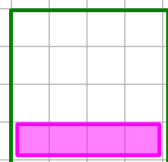
default Map:  
rotation == 0



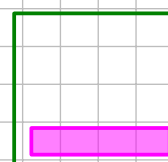
Rotation Map 2:  
rotation == 2



Rotation Map 1:  
rotation == 1



Rotation Map 3:  
rotation == 3



depends on:

Collision Checks:

- with map borders
- with placed bricks

functions:

rotate() → inc rotation; load new Rot Map {0...3}  
 moveRight() → x+1 → block x-coords +1; out of bounds check!  
 moveLeft() → x-1 → block x-coords -1; out of bounds check!  
 moveDown() → y+1 → block y-coords +1; out of bounds check!

Borders:

if (x < 0 || x > width || y > height) {  
 //collide  
}

Placed bricks:

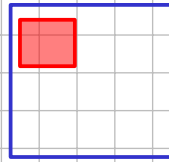
- new position → if any new pos is blocked  
→ //collide

## O-Piece:

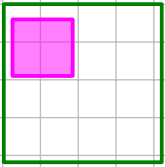
```
O_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

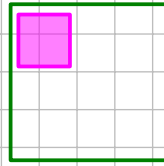
Imp: Default/Map:  
rotation == 0



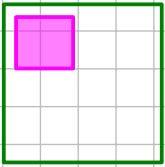
Rotation/Map2:  
rotation == 2



Rotation/Map1:  
rotation == 1



Rotation/Map3:  
rotation == 3

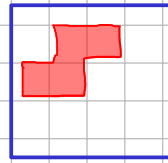


## S-Piece:

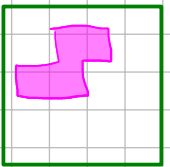
```
S_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

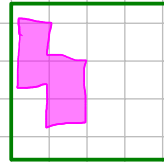
Imp: Default/Map:  
rotation == 0



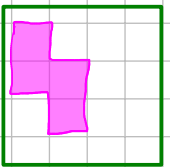
Rotation/Map2:  
rotation == 2



Rotation/Map1:  
rotation == 1



Rotation/Map3:  
rotation == 3

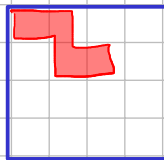


## Z-Piece:

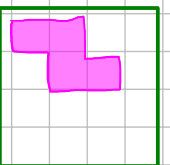
```
Z_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

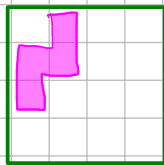
Imp: Default/Map:  
rotation == 0



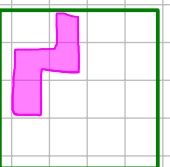
Rotation/Map2:  
rotation == 2



Rotation/Map1:  
rotation == 1



Rotation/Map3:  
rotation == 3

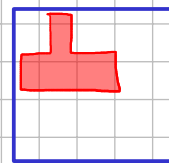


## T-Piece:

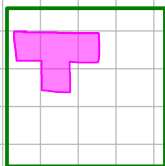
```
T_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

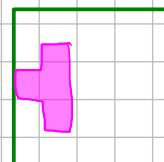
Imp: Default/Map:  
rotation == 0



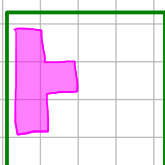
Rotation/Map2:  
rotation == 2



Rotation/Map1:  
rotation == 1



Rotation/Map3:  
rotation == 3

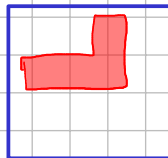


L\_Piece:

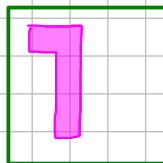
```
L_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

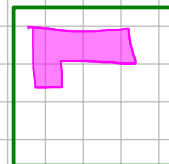
Imp: Default/Map:  
rotation == 0



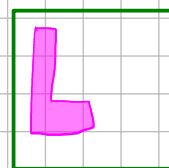
Rotation Map 1:  
rotation == 1



Rotation Map 2:  
rotation == 2



Rotation Map 3:  
rotation == 3

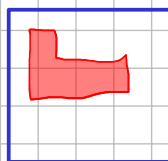


J\_Piece:

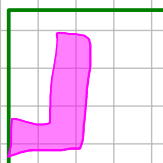
```
J_Piece {  
  x:  
  y:  
  rotation:  
  blocks[]:  
}
```

depends on:

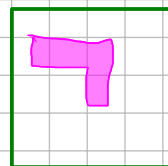
Imp: Default/Map:  
rotation == 0



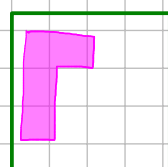
Rotation Map 1:  
rotation == 1



Rotation Map 2:  
rotation == 2



Rotation Map 3:  
rotation == 3



Shape:

```
Shape {  
  type:  
  x:  
  y:  
  blocks[]:  
}
```