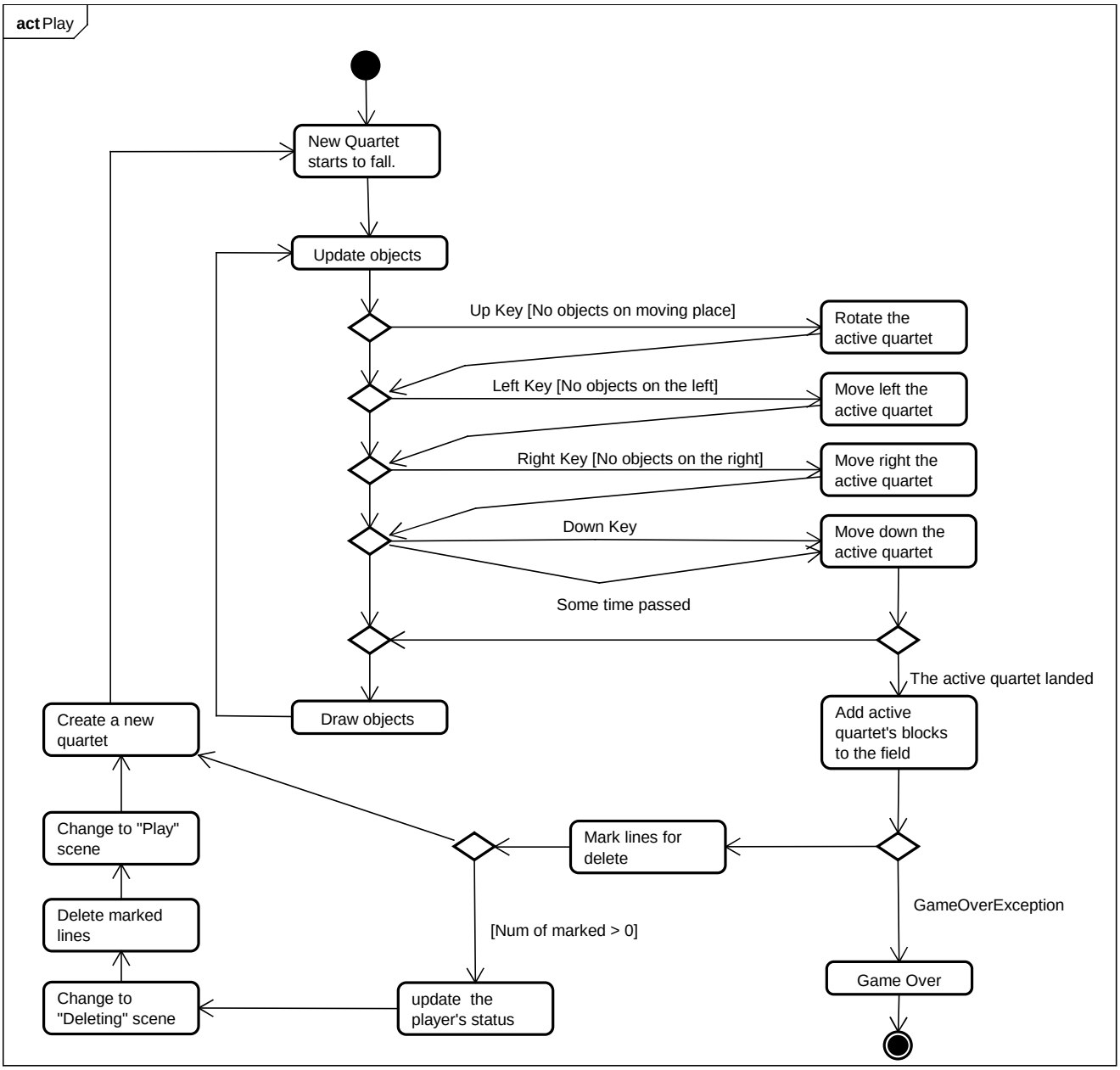
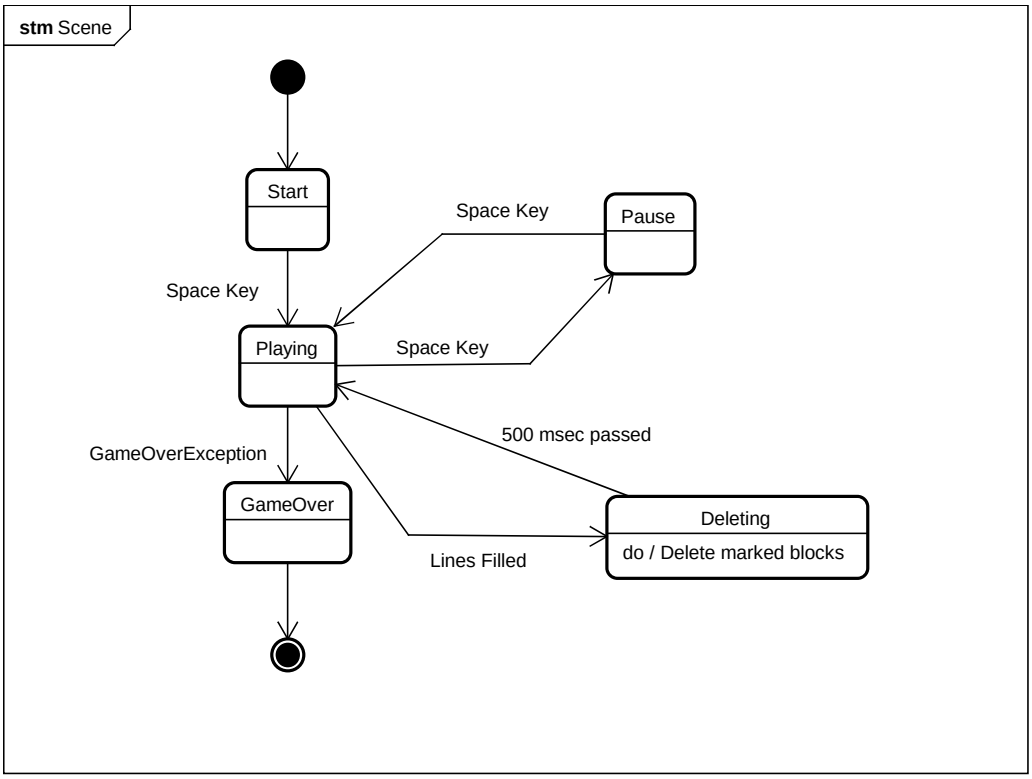
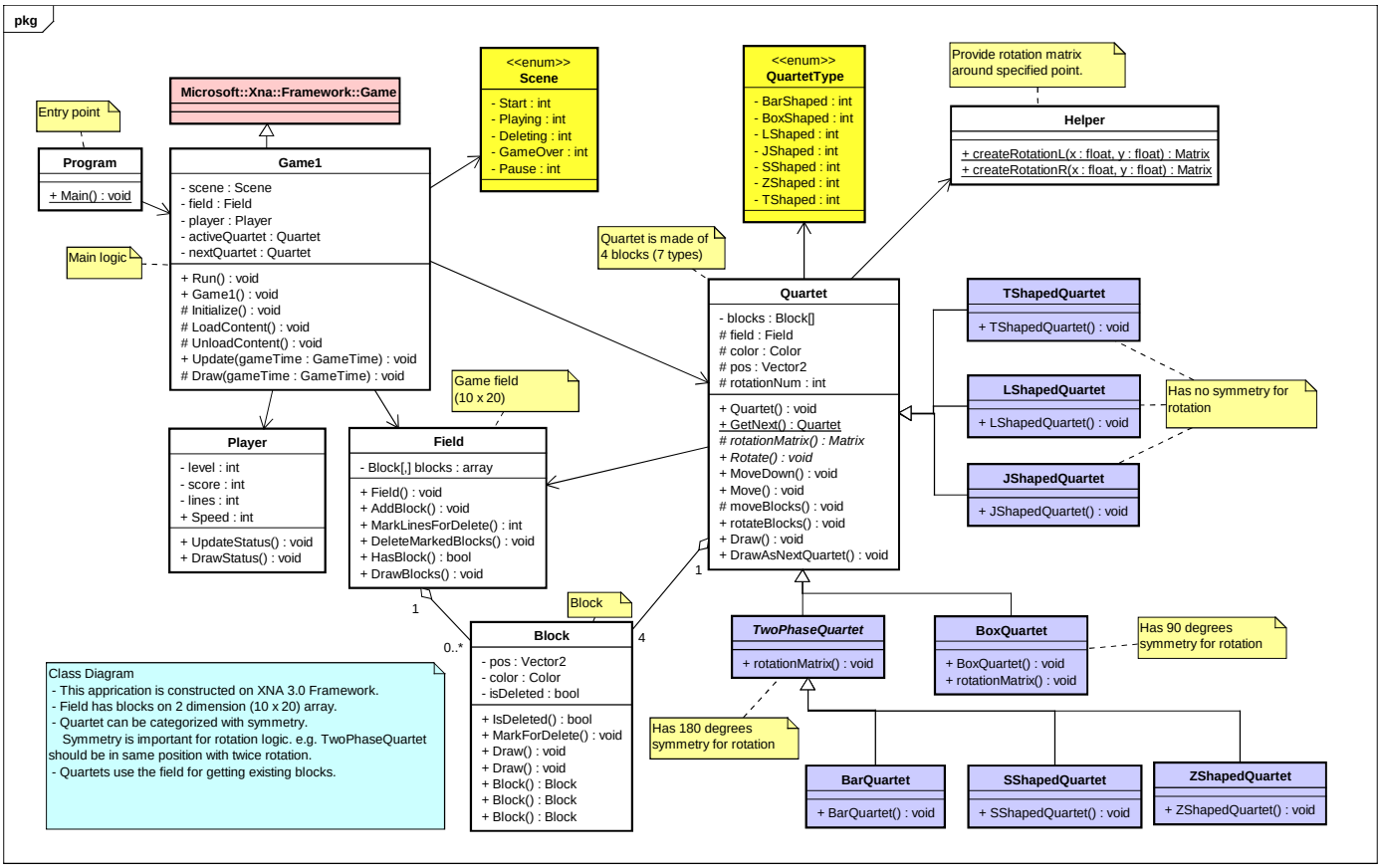

act Play
New Quartet starts to fall.
Update objects
Up Key [No objects on moving place]
Rotate the active quartet
Left Key [No objects on the left]
Move left the active quartet
Right Key [No objects on the right]
Move right the active quartet
Down Key
Move down the active quartet
Some time passed
The active quartet landed
Add active quartet's blocks to the field
Draw objects
Create a new quartet
Change to "Play" scene
Mark lines for delete
GameOverException
Delete marked lines
[Num of marked > 0]
Game Over
Change to "Deleting" scene
update the player's status


stm Scene
Start
Space Key
Pause
Space Key
Playing
Space Key
500 msec passed
GameOverException
GameOver
Deleting
do / Delete marked blocks
Lines Filled


pkg
Entry point
Program
+ Main() : void
Microsoft::Xna::Framework::Game
Game1
- scene : Scene
- field : Field
- player : Player
- activeQuartet : Quartet
- nextQuartet : Quartet
+ Run() : void
+ Game1() : void
# Initialize() : void
# LoadContent() : void
# UnloadContent() : void
+ Update(gameTime : GameTime) : void
# Draw(gameTime : GameTime) : void
Main logic
<<enum>>
Scene
- Start : int
- Playing : int
- Deleting : int
- GameOver : int
- Pause : int
<<enum>>
QuartetType
- BarShaped : int
- BoxShaped : int
- LShaped : int
- JShaped : int
- SShaped : int
- ZShaped : int
- TShaped : int
Provide rotation matrix around specified point.
Helper
+ createRotationL(x : float, y : float) : Matrix
+ createRotationR(x : float, y : float) : Matrix
Quartet is made of 4 blocks (7 types)
Quartet
- blocks : Block[]
field : Field
color : Color
pos : Vector2
rotationNum : int
+ Quartet() : void
+ GetNext() : Quartet
# rotationMatrix() : Matrix
+ Rotate() : void
+ MoveDown() : void
+ Move() : void
# moveBlocks() : void
+ rotateBlocks() : void
+ Draw() : void
+ DrawAsNextQuartet() : void
TShapedQuartet
+ TShapedQuartet() : void
LShapedQuartet
+ LShapedQuartet() : void
Has no symmetry for rotation
JShapedQuartet
+ JShapedQuartet() : void
Game field (10 x 20)
Player
- level : int
- score : int
- lines : int
+ Speed : int
+ UpdateStatus() : void
+ DrawStatus() : void
Field
- Block[,] blocks : array
+ Field() : void
+ AddBlock() : void
+ MarkLinesForDelete() : int
+ DeleteMarkedBlocks() : void
+ HasBlock() : bool
+ DrawBlocks() : void
1
1
0..*
4
Block
Block
- pos : Vector2
- color : Color
- isDeleted : bool
+ IsDeleted() : bool
+ MarkForDelete() : void
+ Draw() : void
+ Draw() : void
+ Block() : Block
+ Block() : Block
+ Block() : Block
TwoPhaseQuartet
+ rotationMatrix() : void
BoxQuartet
+ BoxQuartet() : void
+ rotationMatrix() : void
Has 90 degrees symmetry for rotation
Has 180 degrees symmetry for rotation
BarQuartet
+ BarQuartet() : void
SShapedQuartet
+ SShapedQuartet() : void
ZShapedQuartet
+ ZShapedQuartet() : void
Class Diagram
- This apprication is constructed on XNA 3.0 Framework.
- Field has blocks on 2 dimension (10 x 20) array.
- Quartet can be categorized with symmetry.
Symmetry is important for rotation logic. e.g. TwoPhaseQuartet should be in same position with twice rotation.
- Quartets use the field for getting existing blocks.