

## **FiguresMoving**

- + <a href="mailto:checkAllFunction">checkAllFunction</a>(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkDiagonal(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkPosition(shared\_ptr<Cell>, shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkPositionForKing(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkPositionForKnight(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkPositionForPawn(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- checkVertAndHoriz(shared\_ptr<Cell>, SinglyLinked<Cell>&, SinglyLinked<Cell>&): void
- + erase(shared\_ptr<Cell>): void
- + goTo(shared\_ptr<Cell>, shared\_ptr<Cell>): void

## D raw Board

allotmentSprite: SpriteallotmentTexture: Texture

+ change: bool

changeSprite: Sprite
changeTexture: Texture
enemySprite: Sprite
enemyTexture: Texture
figureSprite: Sprite
figureTexture: Texture
freeCellSprite: Sprite

freeCellTexture: Textureoffset: intsizeOfBoard: int

+ draw(RenderTexture&, int, int, int, shared\_ptr<Cell>, Vector2int, shared\_ptr<SinglyNode<Cell> >, int, shared\_ptr<SinglyNode<Cell> >, int, shared\_ptr<SinglyNode<Cell> >, int, int)

setInformation(int, int): void

-drawBoard/

## Graphic

- backgroundSprite: shared\_ptr<Sprite>
- backgroundTexture: shared\_ptr<Texture>
- drawBoard: DrawBoard
- event: Event
- globalTimer: Clock
- needBackground: bool
- screanHeight: int
- screanWidth: int
- textureForWindow: shared\_ptr<RenderTexture</li>
- window: shared\_ptr<RenderWindow>
- + close(): void
- + draw(Board&, int): void
- + drawInRenderTexture(Board&, int): void
- drawPrivate(): void
- drawPrivate(Board&, int): void
- drawWindow(): void
- + getEvent(): Event
- + getPositionOfMouse(): Vector2int
- + getTimeAsMicroseconds(): long long
- + getTimeAsMilliseconds(): long
- + Graphic(int, int)
- + isOpen(): bool
- + pollEvent(): bool
- + setInformation(int, int, int, int, string): void
- + setInformation(Board&): void