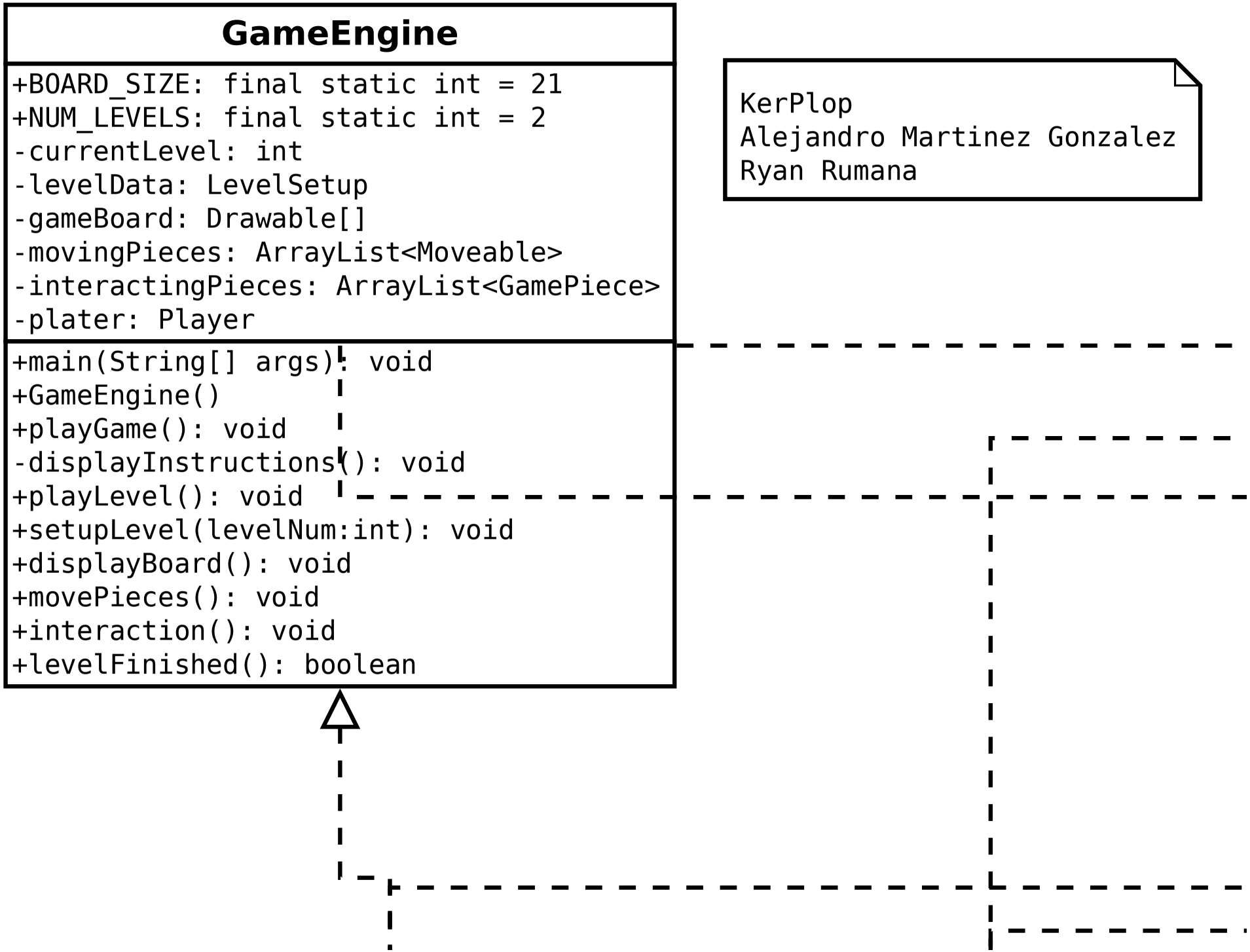
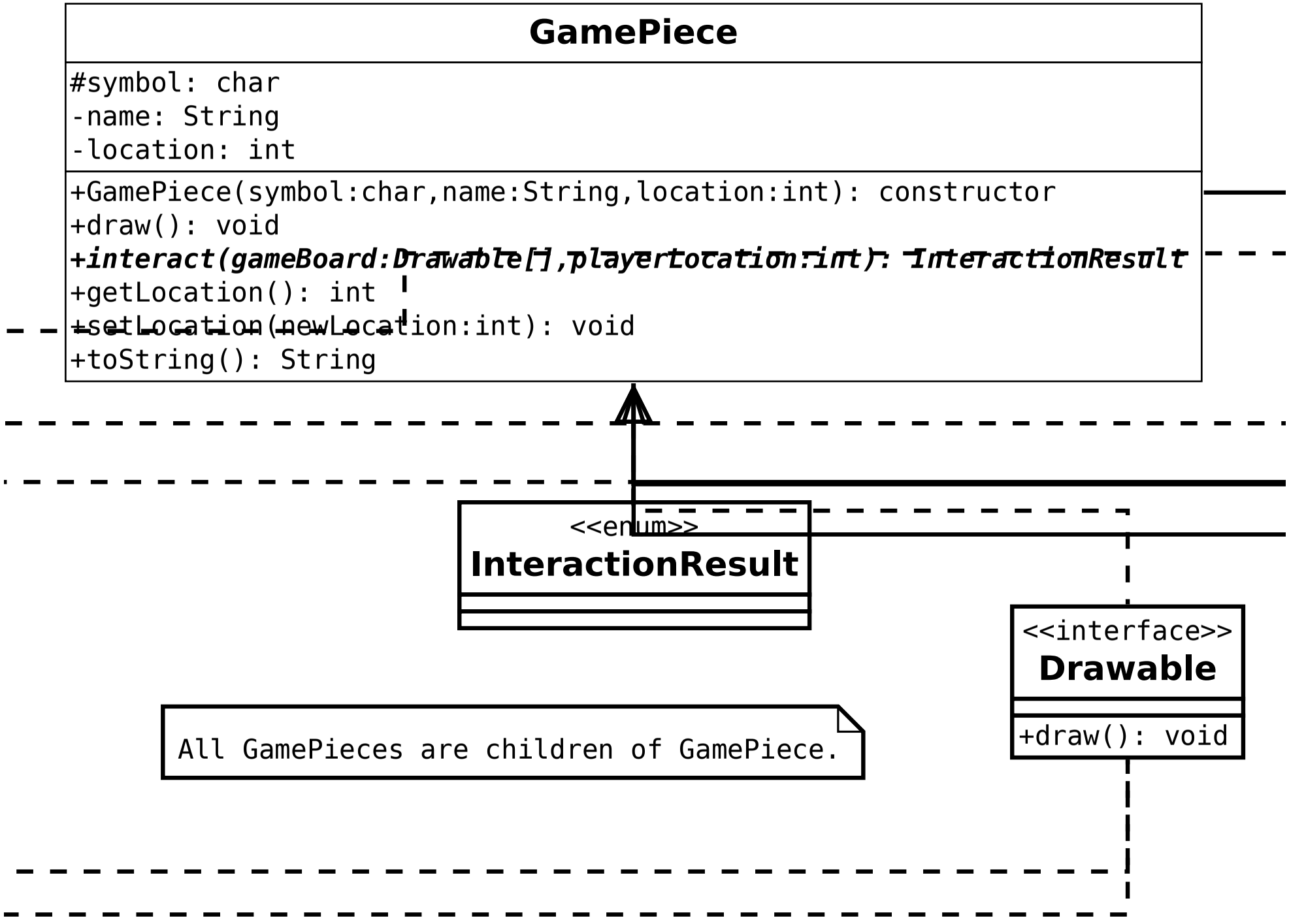
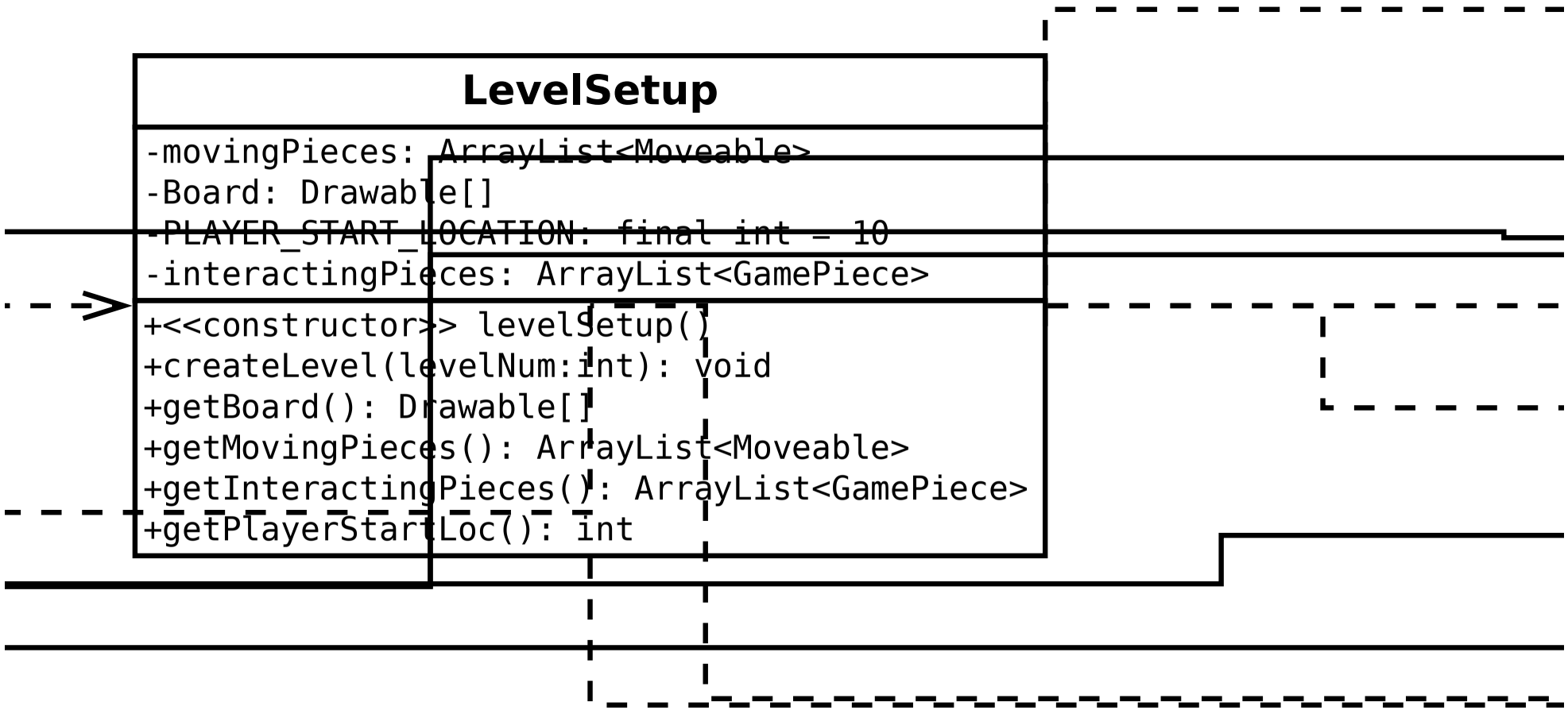
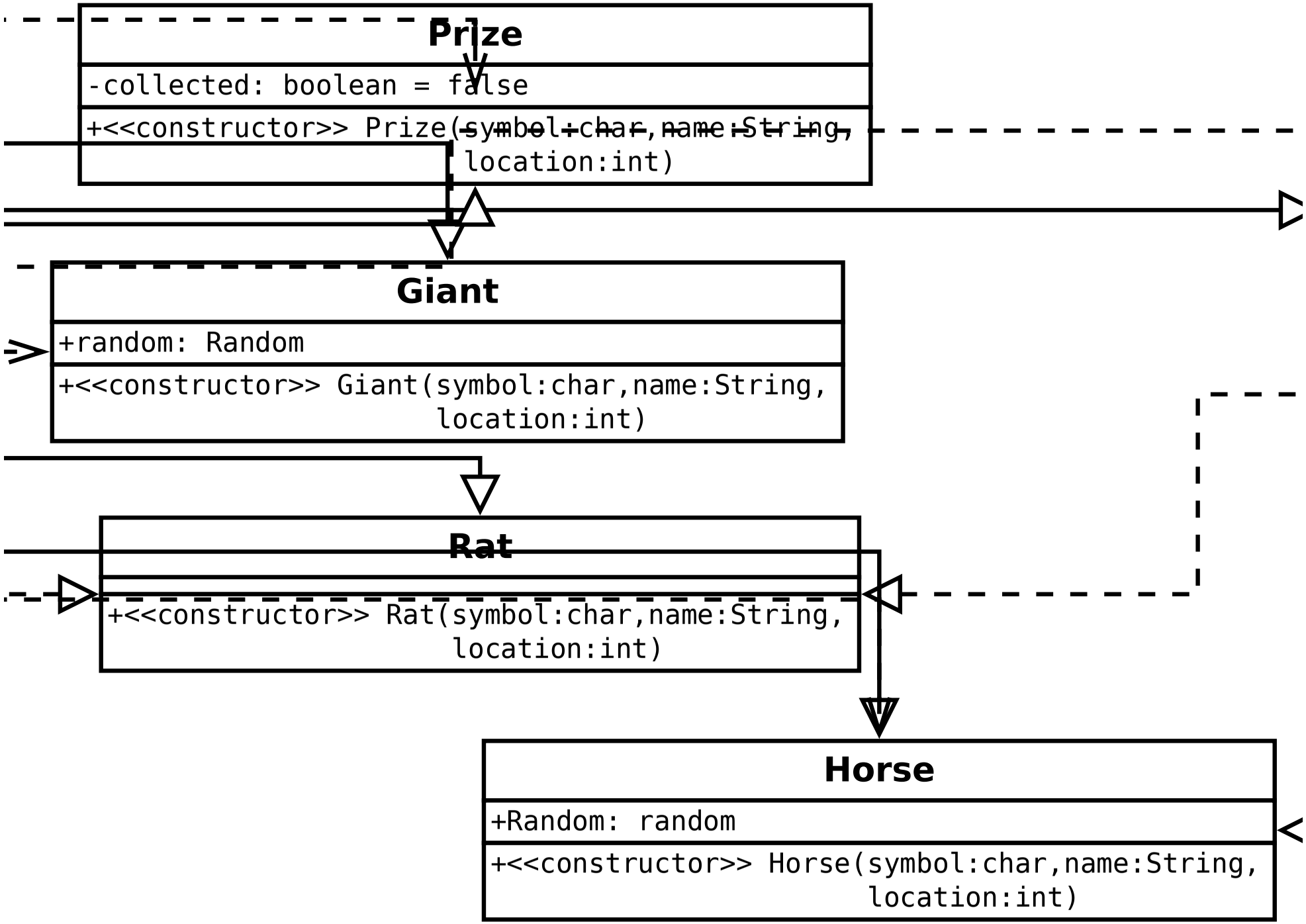
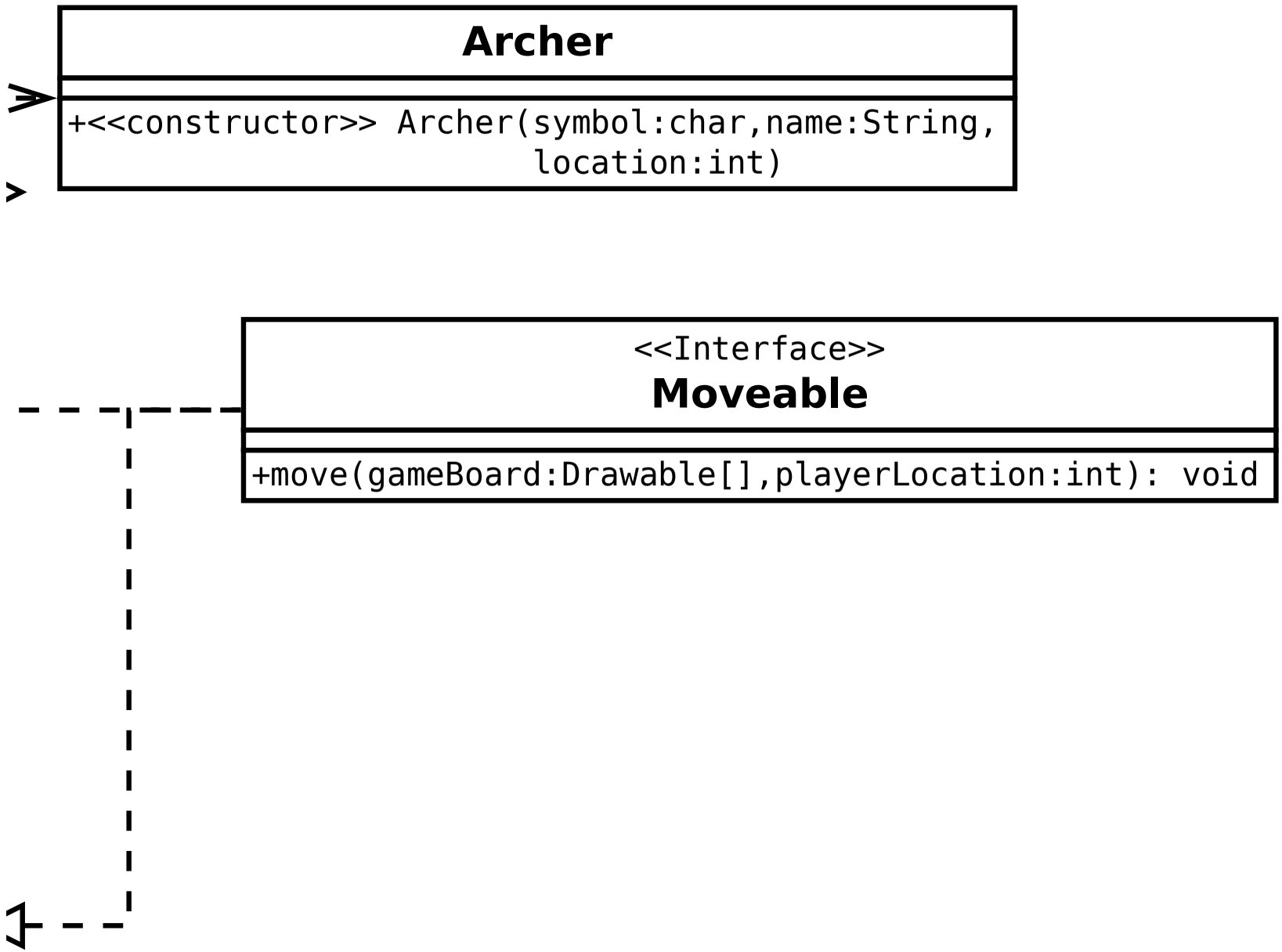
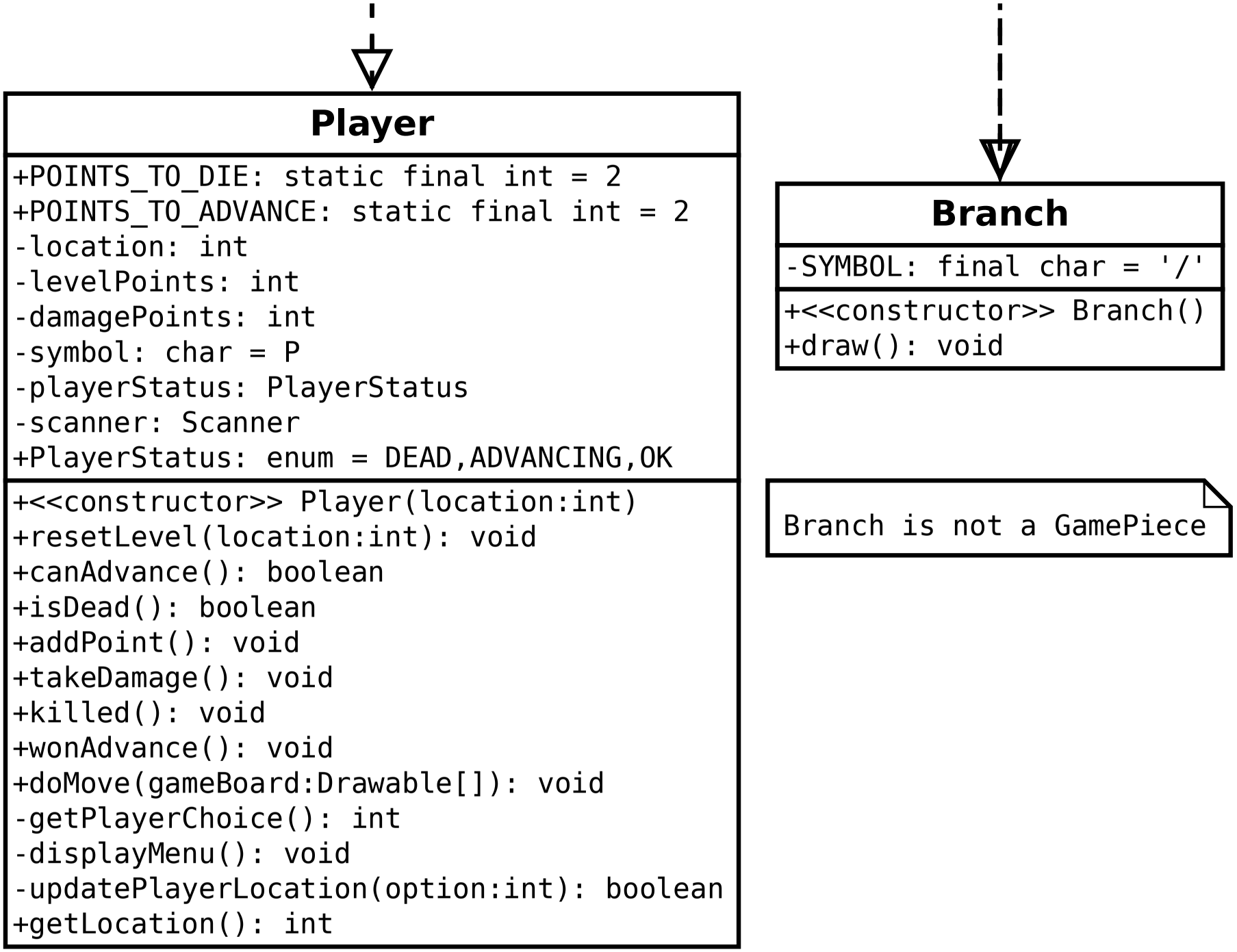

GameEngine
+BOARD_SIZE: final static int = 21
+NUM_LEVELS: final static int = 2
-currentLevel: int
-levelData: LevelSetup
-gameBoard: Drawable[]
-movingPieces: ArrayList<Moveable>
-interactingPieces: ArrayList<GamePiece>
-plater: Player
+main(String[] args): void
+GameEngine()
+playGame(): void
-displayInstructions(): void
+playLevel(): void
+setupLevel(levelNum:int): void
+displayBoard(): void
+movePieces(): void
+interaction(): void
+levelFinished(): boolean
KerPlop
Alejandro Martinez Gonzalez
Ryan Rumana


GamePiece
#symbol: char
-name: String
-location: int
+GamePiece(symbol:char,name:String,location:int): constructor
+draw(): void
+interact(gameBoard:Drawable[],playerLocation:int): InteractionResult
+getLocation(): int
+setLocation(newLocation:int): void
+toString(): String
<<enum>>
InteractionResult
<<interface>>
Drawable
+draw(): void
All GamePieces are children of GamePiece.


LevelSetup
-movingPieces: ArrayList<Moveable>
-Board: Drawable[]
-PLAYER_START_LOCATION: final int = 10
-interactingPieces: ArrayList<GamePiece>
+<<constructor>> levelSetup()
+createLevel(levelNum:int): void
+getBoard(): Drawable[]
+getMovingPieces(): ArrayList<Moveable>
+getInteractingPieces(): ArrayList<GamePiece>
+getPlayerStartLoc(): int


Prize
-collected: boolean = false
+<<constructor>> Prize(symbol:char,name:String,
location:int)
Giant
+random: Random
+<<constructor>> Giant(symbol:char,name:String,
location:int)
Rat
+<<constructor>> Rat(symbol:char,name:String,
location:int)
Horse
+Random: random
+<<constructor>> Horse(symbol:char,name:String,
location:int)


Archer
+<<constructor>> Archer(symbol:char,name:String,
location:int)
<<Interface>>
Moveable
+move(gameBoard:Drawable[],playerLocation:int): void


Player
+POINTS_TO_DIE: static final int = 2
+POINTS_TO_ADVANCE: static final int = 2
-location: int
-levelPoints: int
-damagePoints: int
-symbol: char = P
-playerStatus: PlayerStatus
-scanner: Scanner
+PlayerStatus: enum = DEAD,ADVANCING,OK
+<<constructor>> Player(location:int)
+resetLevel(location:int): void
+canAdvance(): boolean
+isDead(): boolean
+addPoint(): void
+takeDamage(): void
+killed(): void
+wonAdvance(): void
+doMove(gameBoard:Drawable[]): void
-getPlayerChoice(): int
-displayMenu(): void
-updatePlayerLocation(option:int): boolean
+getLocation(): int
Branch
-SYMBOL: final char = '/'
+<<constructor>> Branch()
+draw(): void
Branch is not a GamePiece