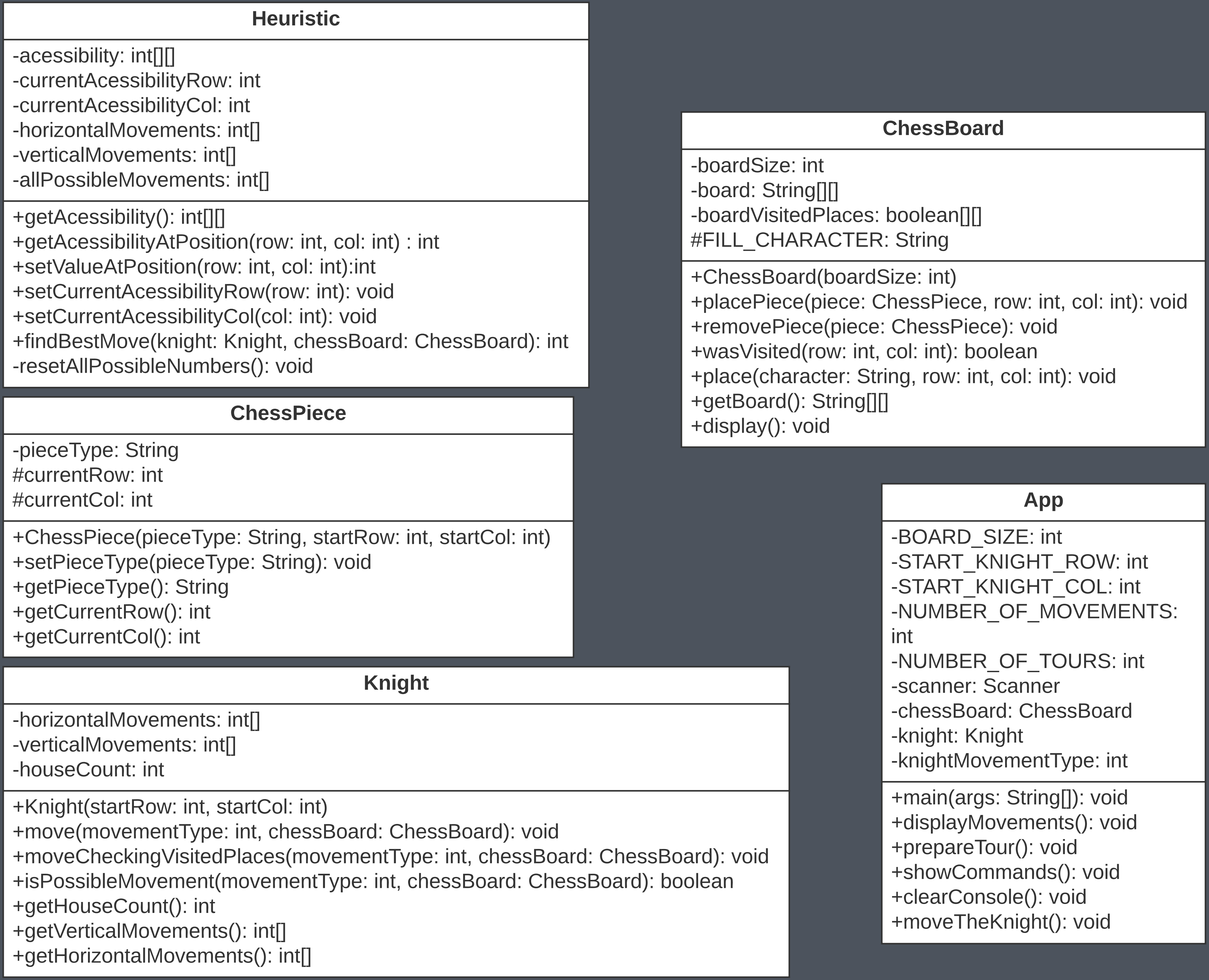
Knight's Tour UML => relations not yet implemented

Arthur Brenno | April 13, 2023



Heuristic

```
-acessibility: int[][]
-currentAcessibilityRow: int
-currentAcessibilityCol: int
-horizontalMovements: int[]
-verticalMovements: int[]
-allPossibleMovements: int[]
```

```
+getAcessibility(): int[][]
+getAcessibilityAtPosition(row: int, col: int) : int
+setValueAtPosition(row: int, col: int):int
+setCurrentAcessibilityRow(row: int): void
+setCurrentAcessibilityCol(col: int): void
+findBestMove(knight: Knight, chessBoard: ChessBoard): int
-resetAllPossibleNumbers(): void
```

ChessPiece

```
-pieceType: String
#currentRow: int
#currentCol: int
```

```
+ChessPiece(pieceType: String, startRow: int, startCol: int)
+setPieceType(pieceType: String): void
+getPieceType(): String
+getCurrentRow(): int
+getCurrentCol(): int
```

Knight

```
-horizontalMovements: int[]
-verticalMovements: int[]
-houseCount: int
```

```
+Knight(startRow: int, startCol: int)
+move(movementType: int, chessBoard: ChessBoard): void
+moveCheckingVisitedPlaces(movementType: int, chessBoard: ChessBoard): void
+isPossibleMovement(movementType: int, chessBoard: ChessBoard): boolean
+getHouseCount(): int
+getVerticalMovements(): int[]
+getHorizontalMovements(): int[]
```

ChessBoard

```
-boardSize: int
-board: String[][]
-boardVisitedPlaces: boolean[][]
#FILL_CHARACTER: String
```

```
+ChessBoard(boardSize: int)
+placePiece(piece: ChessPiece, row: int, col: int): void
+removePiece(piece: ChessPiece): void
+wasVisited(row: int, col: int): boolean
+place(character: String, row: int, col: int): void
+getBoard(): String[][]
+display(): void
```

App

```
-BOARD_SIZE: int
-START_KNIGHT_ROW: int
-START_KNIGHT_COL: int
-NUMBER_OF_MOVEMENTS:
int
-NUMBER_OF_TOURS: int
-scanner: Scanner
-chessBoard: ChessBoard
-knight: Knight
-knightMovementType: int
```

```
+main(args: String[]): void
+displayMovements(): void
+prepareTour(): void
+showCommands(): void
+clearConsole(): void
+moveTheKnight(): void
```