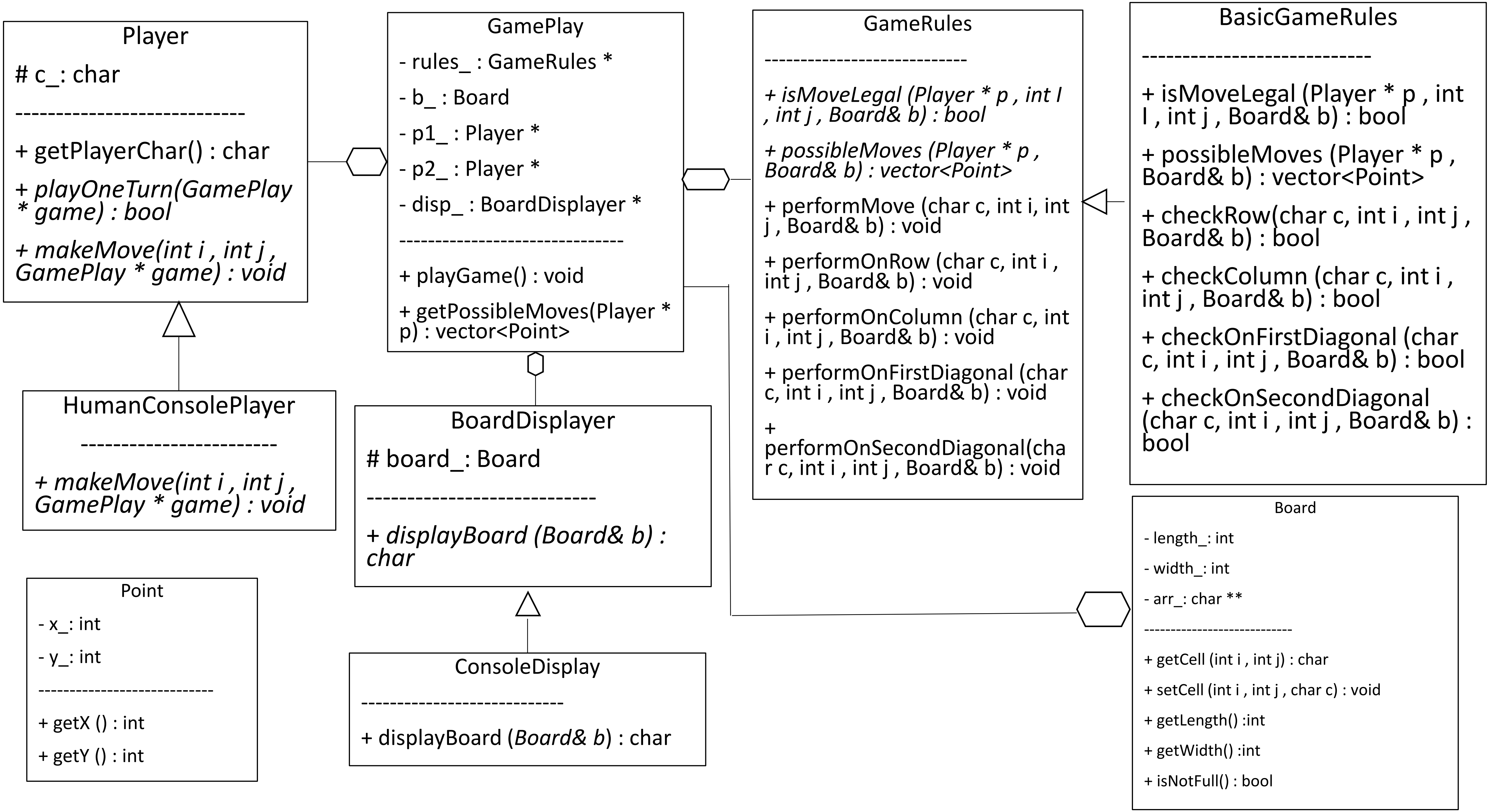Player
# c_: char

+ getPlayerChar() : char
*+ playOneTurn(GamePlay * game) : bool*
*+ makeMove(int i , int j , GamePlay * game) : void*
GamePlay
- rules_ : GameRules *
- b_ : Board
- p1_ : Player *
- p2_ : Player *
- disp_ : BoardDisplayer *

+ playGame() : void
+ getPossibleMoves(Player * p) : vector<Point>
GameRules
----
*+ isMoveLegal (Player * p , int I , int j , Board& b) : bool*
*+ possibleMoves (Player * p , Board& b) : vector<Point>*
+ performMove (char c, int i, int j , Board& b) : void
+ performOnRow (char c, int i , int j , Board& b) : void
+ performOnColumn (char c, int i , int j , Board& b) : void
+ performOnFirstDiagonal (char c, int i , int j , Board& b) : void
+ performOnSecondDiagonal(char c, int i , int j , Board& b) : void
BasicGameRules

+ isMoveLegal (Player * p , int I , int j , Board& b) : bool
+ possibleMoves (Player * p , Board& b) : vector<Point>
+ checkRow(char c, int i , int j , Board& b) : bool
+ checkColumn (char c, int i , int j , Board& b) : bool
+ checkOnFirstDiagonal (char c, int i , int j , Board& b) : bool
+ checkOnSecondDiagonal (char c, int i , int j , Board& b) : bool
HumanConsolePlayer
----
*+ makeMove(int i , int j , GamePlay * game) : void*
BoardDisplayer
# board_: Board

+ displayBoard (Board& b) : char
Board
- length_: int
- width_: int
- arr_: char **

+ getCell (int i , int j) : char
+ setCell (int i , int j , char c) : void
+ getLength() :int
+ getWidth() :int
+ isNotFull() : bool
Point
- x_: int
- y_: int

+ getX () : int
+ getY () : int
ConsoleDisplay

+ displayBoard (*Board& b*) : char