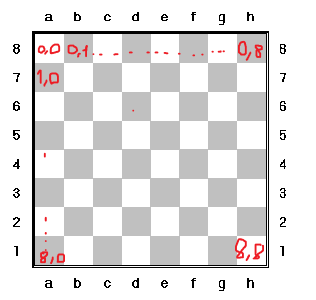
מסמך אפיון למשחק השחמט

נכתוב את התוכנה בJS לקוח ושרת: לקוח טריוויאלי ושרת בnodejs.

מחזיקי נתונים – לא חייבים מחלקות, יתכן וJSON מספיק

מחלקה position

מחלקה של אורך ורוחב שתתאר משבצת ספציפית בלוח.  
- שני שדות מספריים של row, col.

* תמיכה במיפוי מתצוגה של אותיות/מספרים לשני מספרים. כוונה:
* 

אפשר לדעת ש “b7” זה 1,1 בשדה הrow והcol. "a7" זה 1 בשדה הrow ו0 בcol וכו'.

* תמיכה בחיבור וחיסור בין position שונים (לצורך וקטור כיוונים שנדון עליו אחכ)

מחלקה move:

תורכב משני שדות של position שהם יהיו הsrc והdst. תייצג מעבר בין שני משבצות. מחלקה לנוחות בלבד לסידור נתונים ולא מייצגת משהו לוגי מורכב.

מחלקת Board

מחלקה שמחזיקה 10 על 10 PIECE . זה מכיוון שאנחנו נרפד את הלוח לצורך יעילות בשביל לחסוך בבדיקות תקינות של פוזישנים.

מחלקה ENUM של BLACK WHITE

לוגיקה

צורת החשיבה השחמטאית לתזוזה של חיילים היא בתצורה הבאה: בהינתן חייל, המיקום שלו והלוח בו הוא נמצא ניתן לדעת:  
- משבצות שהוא יכול לזוז אליהן  
- משבצות שהוא יכול לאכול.

שתי הפונקציונאליות האלו מתממשות לאור כל השלושה כאשר השוני הוא סוג החייל: כלומר המימוש שונה בהתאם לסוג החייל ולכן מתאימה לפה תכונה של ירושה ממחלקה אחת אבסטרקטית.

מחלקת piece

כמו שציינו: השדות:רפרנס ללוח ומיקום נוכחי (position) . כמו כן לשם נוחות נוסיף את צבע הכלי (שחור/לבן) שתי מתודות אבסטרקטיות של אופציות תזוזה ואופציות אכילה.

ממנה ירשו כל הכלים השונים: PAWN,ROOK,KNIGHT,BISHOp,QUEEN,KING  
ככל הנראה נצטרך את PAWN להפריד גם כן לשני מחלקות של White Pawn ו Black Pawn שכן מדובר בכיווני תזוזה שונים.

בנוסף תירש ממנה מחלקה של BLANK שהיא חייל ריפוד.

מחלקה Game

תכיל שדה Board והיא תנהל את המשחק.

תכיל פונקציה שמאפסת את הלוח.

תדע של מי התור הנוכחי.