**תיעוד החלטות בבניית פרויקט משותף**

Avshalom Tam 203829478 & Steve Gutfreund EM168887

**שיקולי OOP**

משיקולי קוד מונחה עצמים, החלטנו לקחת את מבנה הפרויקט של סטיב. הסיבה לכך היא כיוון שאצלנו ישנה מחלקה בשם "GameRules”. מחלקה זו יורשת ממחלקת “GameLogic” שהינה מחלקת אב וירטואלית לגמרי. וכך נוכל לתמוך בשינוי כללי המשחק.

תפקידה של “GameLogic” הוא להוות מחלקת על לכל פונקציות לוגיקת המשחק הבסיסיות של המשחק Reversi ללא תלות בקביעת חוקים כאלה או אחרים של המשחק. לשם ההבהרה, הפונקציות שב “GameLogic” תקיפות בין אם לשחקנים מותר לשים אריח שלהם בתוך משבצת רק במידה והם הופכים אריחים של השחקן השני ובין אם לאו.

לעומתה, המחלקה היורשת "GameRules” מגדירה את החוקים של המשחק בווריאציה הספציפית שלו כך שניתן להשתמש בה או בכל מחלקה אחרת שיורשת מ“GameLogic” על מנת לשחק.

**שיקולי מימוש של פונקציות**

משיקולי מימוש, בפונקציה הבודקת את תקינות הקלט שהכניס המשתמש במידה והוא מסוג "HumanPlayer”, החלטנו לקחת את המימוש של אבשלום. הסיבה לכך היא שפונקציה זו הבדילה בהתרעת המשתמש על קלטים לא חוקיים. כלומר, במידה והמשתמש הכניס קלטים לא חוקיים בגלל גודלם בלבד,שהוא מעבר לגבולות הלוח, המערכת תזרוק exception ותחזיר לו שגיאה מתאימה והיא "!Out of the game boarders” אחרת, במידה והשחקן הכניס קלט לא תקין בעליל, המערכת תזרוק exception ותחזיר לו שגיאה מתאימה “wrong parameters”.

**שיקולי מימוש של פונקציית הmain**

משיקול של תפקיד פונקציית הmain שיוצרת מופע של המחלקה Game, מאתחלת משחק ומריצה אותו, החלטנו לקחת את המימוש של סטיב. במימוש זה, יצירת הBoard, ויצירת השחקנים נעשית בתוך פונקציית initialize של game, ולא במיין עצמו. כלומר, מכיוון שתפקיד הmain הוא תיאור מכליל מאוד של המשחק ללא ירידה לפרטים התכנים, הוא אמור להיות קצר ביותר ולכלול רק את איתחול המשחק והרצתו ע"י פונקציית run. כך מתקבלת פונקציה קצרה וברורה בעלת 4 שורות קוד בלבד (שהאחרונה מהן היא return 0).