**חלק א'**

3. השחקן ReflexPlayer בודק תחילה מהם המהלכים החוקיים מהמצב שבו הוא נמצא(חסימה מבחינת קיר או גבולות הלוח) ומשערך את כל אחד מן המצבים הללו ובוחר בצורה אקראית את אחד מבין הטובים ביותר. השחקן משערך מצב לפי הניקוד שיקבל אילו היה מבצע מהלך זה. למשל, עבור הפסד היה מאבד 500 נק', עבור ניצחון מרוויח 500 וכו'.

**חלק ב'**

1. TODO
2. TODO

**חלק ג'**

1. מאחר ויצרנו עץ שתורות השחקנים בו הם לפי שכבות, ההנחה שלנו היא שהשחקנים משחקים לפי סדר קבוע כלשהו. בצורה שבה מומש המשחק בקובץ pacman.py ההנחה מתקיימת מאחר והשחקנים משחקים לפי התור בלולאת המשחק המרכזית. במשחקים האחרים אולי הנחה זו לא תתקיים, בין אם הסדר משתנה או בין אם התורות אינם סנכרונים.

3. דרך נוספת למימוש האלגוריתם שלא דורשת פיתוח שכבה נוספת עבור כל רוח, הינה לפתח שכבת מינימום עבור הרוח הכי קרובה לפקמן. (קרובה למשל ע"פ מרחק מנהטן).

חסרונות: ראיית משחק מוגבלת – לא מתחשבים בכל מהלכי הרוחות האפשריים אלא רק בזו שקרובה ביותר אלינו.

יתרונות: חישוב ערך הMinMax באופן מהיר יותר כיוון שכעת עלינו לפתח רק שכבה אחת בלבד בכל עומק במקום כמספר הרוחות.

**חלק ד'**

1. מבנה העץ החדש שהגדרנו אכן משפיע על האלגוריתם בכך שבמקרה שיש שכבת MIN מתחת לשכבת MIN במידה והשכבה התחתונה שולחת -אינסוף במקרה של גיזום, אז ניתן להעביר את הערך הזה כלפי מעלה (כלומר גיזום עמוק יותר ) בשונה משליחת -אינסוף לשכבת מקסימום מעל שרק תגזום את הענף הנתון בלבד ואולי לא תבצע גיזום גדול יותר כמו בשתי שכבות הMIN.